

HOMOTOPY MODEL THEORY I: SYNTAX AND SEMANTICS

DIMITRIS TSEMENTZIS

ABSTRACT. A model theory in the framework of Univalent Foundations requires a logic that allows us to define structures on homotopy (n -)types, similar to how first-order logic can define structures on sets. We define such an “ n -level” logic for finite n . The syntax is based on a generalization of Makkai’s FOLDS, obtained by an operation that allows us to add equality sorts to FOLDS-signatures. We then give both a set-theoretic and a homotopy type-theoretic semantics for this logic and prove soundness for both with respect to an appropriate deductive system. As an application, we prove that univalent categories are axiomatizable in 1-logic.

CONTENTS

Introduction	1
1. Preliminaries	4
2. Syntax of n -logic	8
3. Homotopy Semantics of n -logic	11
4. Set-Theoretic semantics for 1-logic	16
5. Proof System for n -logic	22
6. Soundness	25
7. Examples and Applications	29
Appendix: Proof System	36
References	37

INTRODUCTION

The Univalent Foundations of Mathematics (UF) [Uni13] take their basic objects to be homotopy types. In UF mathematical structures are therefore encoded as structured homotopy types, similar to how in set-theoretic foundations they are encoded as structured sets. This basic picture allows us to envision a model theory in which the basic syntax no longer describes structured sets, but structured homotopy types. If “set-theoretic model theory” is understood as model theory in set-theoretic foundations then “homotopy type-theoretic model theory” would be the corresponding

Date: March 11, 2016.

2010 Mathematics Subject Classification. 03G99, 03B15, 03B22, 03C99.

Key words and phrases. Univalent Foundations, Categorical Logic, Homotopy Type Theory.

model theory in the Univalent Foundations. The crucial technical constraint of having the title fit in a single line forced us to compromise for “Homotopy Model Theory”. The aim of the present series of papers is to develop this model theory.

In [CK90] model theory is defined as “the branch of mathematical logic which deals with the relation between a formal language and its interpretations, or models.” Therefore, in order to do any kind of model theory we must at least have a formal language (*syntax*) and an interpretation of that formal language (*semantics*). This first paper will develop these two components. They comprise what we will call “ n -logic” (for $-1 \leq n < \infty$) in order to reflect the following fundamental (and revolutionary) idea of UF: an “ n -level” theory is a piece of syntax whose models consist of structures defined on n -groupoids/homotopy n -types.

The syntax of n -logic will be based on a suitable extension of the syntax of Makkai’s FOLDS (First-Order Logic with Dependent Sorts) as it was developed in [Mak95]. A fundamental insight of Makkai was that the syntax of “higher-level” theories can itself be presented as a category. In particular, the signatures of FOLDS can be described as one-way categories where the arrows encode variable dependencies between the objects (understood as “sorts”). The key in defining n -logic is to “add equalities” to these FOLDS signatures. Indeed, it is helpful to think of n -logic as standing to FOLDS in the same relation that first-order logic with equality stands to first-order logic. To add equality to a first-order signature Σ , one simply adds a binary relation with a certain fixed denotation. The analogous process for FOLDS is carried out in terms of a “globular completion” operation on categories which attaches (globular towers of) “equality sorts” to pre-existing sorts in \mathcal{L} in a manner compatible with their “height”. Roughly speaking, the signatures of n -logic are then the “globularly completed” FOLDS signatures of Makkai. This marks the first main contribution of this paper: a general definition of an “ n -level” syntax.

The most natural semantics for n -logic is in UF. The basic idea of these “homotopy semantics” is the following: non-logical sorts of dimension $m \leq n$ are interpreted as (dependent functions landing in) homotopy m -types, as the latter are formalized in (some) homotopy type theory (HoTT). The equality sorts that have been added to the syntax through globular completion are then interpreted as identity types. Although we will describe these semantics using the formal notation of dependent type theories (Π , Σ , Id-types etc.) we stress that no such fixed formal system is needed in order to make sense of the interpretation at an intuitive level. (The situation is entirely analogous to the set-theoretic semantics of first-order logic which can be understood at an intuitive level independent of any specific choice of formalism (e.g. ZFC, NBG) for the ambient set theory.) This marks the second main contribution of this paper: a semantics for n -logic that can be used to define structures in any HoTT.¹

The semantics of n -logic can also be defined in a set-theoretic metatheory for suitable set-theoretic notions of n -groupoids/homotopy n -types. We will thus also define a set-theoretic semantics for 1-logic based on the more traditional notion of a groupoid

¹For the purposes of this paper, HoTT will refer to the system outlined in [Uni13], i.e. intensional MLTT with at least one univalent universe. But the semantics can easily be modified to accommodate alternative HoTTs, e.g. [CCHM15].

understood as a category whose every morphism is invertible. These semantics interpret extensions of formulas in 1-logic as pseudonatural transformations from contexts to \mathcal{L} -structures. This marks the third main contribution of this paper: a groupoid semantics for a dependently-typed syntax entirely independent of the machinery of contextual/comprehension/type categories or C -systems [Car86, Jac99, Voe14] as well of the original groupoid interpretation of Hofmann and Streicher [HS98].

A natural question now is whether there is a proof system on the syntax of n -logic that can justify its homotopy semantics. In other words, even though the operation of globular completion adds “equality sorts” can we make their variables behave like paths in such a way that justifies their (fixed) denotation? We present such a proof system for n -logic. It is based on a standard sequent calculus for first-order logic to which we add three rules: two rules governing the existence and uniqueness of the “reflexivity” paths of equality sorts and a propositional version of the J-rule. Since the syntax of n -logic is purely relational (no closed terms) these assume an unfamiliar form, but they are in spirit very much related to the rules for identity types of MLTT [ML84]. This system is then shown to be sound with respect to both the above-sketched semantics, which marks the fourth main contribution of this paper.

This is the first of a projected three papers on n -logic. Part II will be devoted to completeness theorems for both the semantics and proof system outlined in this paper for the case $n = 1$. Part III will focus on the problem of the definability of (semi-)simplicial types.

Related Work. There are two main reasons we are interested in developing the model theory outlined above. Both relate to ongoing work in UF. Firstly, our framework provides a general definition of a signature for structures definable in the framework of UF. It could thus be used to generalize the Structure Identity Principle of [Uni13]. Similar work has been carried out by Shulman, North and Ahrens [ANS14] who consider FOLDS in its capacity to provide a general notion of isomorphism for higher categories. Formalizations of category theory in the style of FOLDS has also been carried out by Ahrens under the UniMath project [VAG⁺] and his formalization overlaps with some of the material of Section 7. Secondly, we intend n -logic as a tool for the study of homotopy type theories as mathematical objects themselves in such a way that can itself be formalized inside UF. In other words, n -logic can be used as a framework for doing metamathematics “natively” in UF, independent of any ambient set theory. This relates to ongoing work on the “Initiality Conjecture” in the series of papers by Voevodsky beginning with [Voe14] (except we note that for its specific purposes this work takes place in ZF set theory). It also relates to work on “internal setoids” [PW14] which is focused on implementing tools to carry out the metatheory of type theories inside type theory.

Outline of the Paper. In Section 1 we introduce the syntax of FOLDS. In Section 2 we introduce the operation of globular completion (Definition 2.1) and use it to define the syntax of n -logic for each (finite) n . In Section 3 we define the homotopy theoretic semantics of n -logic and in Section 4 we define a set-theoretic

semantics for 1-logic. In Section 5 we define a proof system for n -logic and then in Section 6 prove soundness for both our semantics with respect to that proof system. Finally, in Section 7 we consider precategories as a FOLDS theory and then apply our framework to prove that univalent categories are “1-elementary” in the sense that they are axiomatizable by a 1-theory (Proposition 7.3).

Acknowledgements. I thank Benedikt Ahrens, John Burgess, Chris Kapulkin, Peter Lumsdaine, Anders Mörtberg, Paige North, Dan Grayson and Vladimir Voevodsky for helpful and stimulating conversations and emails.

1. PRELIMINARIES

We will assume familiarity with the basics of categorical logic as well as of intensional Martin-Löf Type Theory and its homotopy interpretation. For category-theoretic background [ML98] remains the standard reference; for the basics of dependent type theory we recommend [ML84, Hof97]; for the homotopy interpretation and an introduction to the Univalent Foundations see [Uni13, KLV14] and references therein.

We will now present in more detail the basic syntax of FOLDS following closely Makkai’s original presentation in [Mak95]. Our presentation will be given in “functorial” rather than “inductive” style. The apparent problem with the “functorial” style is that it invokes machinery much too strong to deserve the name “syntax”. However, all concepts introduced here can in fact be defined recursively with minimal assumptions since we impose appropriate finiteness conditions throughout.

Definition 1.1 (FOLDS signature). A (*finite*) *FOLDS-signature* is a category \mathcal{L} with $|\text{ob}\mathcal{L}| < \aleph_0$ and $|\text{mor}\mathcal{L}| < \aleph_0$ together with a grading

$$d: \text{ob}\mathcal{L} \rightarrow \mathbb{Z}_{\geq -2}$$

such that for any $f: K \rightarrow K_f$ with $f \neq 1_K$ we have $d(K_f) > d(K)$ (where we write K_f for the codomain of f). We call $d(K)$ the *dimension* of K . We define the *height* of \mathcal{L} as

$$h(\mathcal{L}, d) = \sup_{K \in \text{ob}\mathcal{L}} d(K)$$

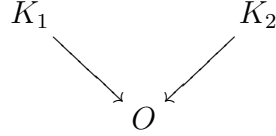
The dimension grading d induces (canonically) a stratification of \mathcal{L} into levels, defined (inductively) as follows:

$$l(K) = \begin{cases} 0 & \text{if } K \text{ is the domain only of } 1_K \\ \sup_{f: K \rightarrow K_f} l(K_f) + 1 & \text{otherwise} \end{cases}$$

We will call $l(K)$ the *level* of K .

Remark 1.2. It is not hard to see that Definition 1.1 is a generalization of Makkai’s definition of a FOLDS signature in [Mak95]. In particular, the grading d ensures that \mathcal{L} is a finite, one-way and reverse well-founded category. Such categories are also known as *inverse* categories, except we restrict ourselves to finite ones. The dimension d is extra structure not present in the original definition.

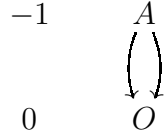
Remark 1.3. It is important to note that even though the level function is uniquely induced by d , it is possible to have objects that are of the same level, but of different dimension. For example consider \mathcal{L} given by



with $d(K_1) = 2$, $d(K_2) = 3$ and $d(O) = 5$.

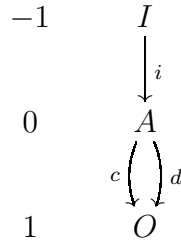
We will sometimes call an object R of maximal level a *relation* and we call the rest of the objects of \mathcal{L} *sorts*. (We will usually reserve the letter R for the former and the letter K for the latter.)

Example 1.4. Let $(\mathcal{L}_{\text{graph}}, d)$ denote the following FOLDS signature, with the numbers on the left representing the dimension of the corresponding sorts.



As the name suggests this would be the signature in which to talk about graphs, where A would encode the “edges” between previously declared “vertices” of sort O . We will usually omit explicit mention of d , and write simply $\mathcal{L}_{\text{graph}}$ for the above signature.

Example 1.5. Similarly, omitting d , we let \mathcal{L}_{rg} denote the following signature



subject to the relation $di = ci$. Intuitively, this corresponds to the signature for reflexive graphs, where I is a unary predicate that can only be “asked” of an “arrow” in A that we already know is a loop. \mathcal{L}_{rg} will serve as a fundamental example since it represents the simplest case sufficient to illustrate most of the concepts we are going to introduce.

Another notion that is crucial for us is that of a (finite) extension of one FOLDS-signature by another.

Definition 1.6 (Extension of a signature). Let (\mathcal{L}, d) be a FOLDS signature. Then a FOLDS signature (\mathcal{L}', d') is an *extension* of (\mathcal{L}, d) iff $\text{ob}\mathcal{L}' \supset \text{ob}\mathcal{L}$, $\text{mor}\mathcal{L}' \supset \text{mor}\mathcal{L}$, \mathcal{L} is a full subcategory of \mathcal{L}' and $d'|_{\text{ob}\mathcal{L}} = d$. We write $(\mathcal{L}', d') > (\mathcal{L}, d)$ to indicate that (\mathcal{L}', d') is an extension of (\mathcal{L}, d) .

For the rest of this section we assume we are given a FOLDS signature (\mathcal{L}, d) . We will now define a language out of it. By a *language* here we mean the collection of all concepts standardly associated with a syntax: variables, formulas, sequents and rules for determining the well-formedness of each. (At this point the dimension d plays no essential role other than inducing the level function l .)

Definition 1.7 (Variables and Contexts). *Variables* are given by a functor

$$V: \mathcal{L} \rightarrow \mathbf{Set}$$

satisfying the following conditions:

- (1) $V(K) \cap V(K') = \emptyset$ for $K \neq K' \in \text{ob}\mathcal{L}$
- (2) $|V(K)| = \aleph_0$ for all $K \in \text{ob}\mathcal{L}$
- (3) For every finite (i.e. finite sets in all its values) subfunctor $\Gamma \subset V$ we have

$$|\{y \mid \text{dep}(y) \subset \Gamma\}| = \aleph_0$$

where

$$\text{dep}(y) =_{\text{def}} \{V(f)(x) \mid \text{dom}(f) = K\}$$

is the set of *dependent variables* of x and

$$|\Gamma| = \bigcup_{K \in \mathcal{L}} \Gamma(K)$$

We write $x: K$ as an abbreviation for $x \in V(K)$. A *context* is a finite subfunctor of V and given two such contexts Γ, Δ a *context morphism* is a natural transformation between them. For any two contexts Γ and Δ we write $\Gamma \cup \Delta$ for their union as subobjects of V , i.e. the functor that takes $K \mapsto \Gamma(K) \cup \Delta(K)$. If Γ is a context and $x: K$ a variable such that $\text{dep}(x) \subset \Gamma$ then we write $\Gamma, x: K$ for the context that is the same as Γ everywhere except $\Gamma, x: K(K) = \Gamma(K) \cup \{x\}$. Like with contexts we write We write

$$|V| = \bigcup_{K \in \mathcal{L}} V(K)$$

For a given context Γ we write

$$\Gamma^\uparrow = \{y \in |V| \mid y \notin \text{dep}(x) \forall x \in |\Gamma|\}$$

When $x \in \Gamma^\uparrow$ and $x \notin \Gamma$ we say that x is *fresh* for Γ .

Remark 1.8. Condition (3) in Definition 1.7 is there to ensure that there are always enough variables available to us given any choice of other variables that they may depend on. Since the notions we use to express condition (3) are also introduced in terms of V there is clearly a circularity here. But it is one that can easily be circumvented with a mutually inductive definition. How to actually implement such a definition is of course a non-trivial technical problem.

It is easy to check that the notions in Definition 1.7 encode the same amount of information as the corresponding notions in MLTT. We will usually write out contexts in the MLTT style. We will also often abbreviate contexts by writing out simply the names of variables in them.

Example 1.9. Let $f \in V(A)$ and $x, y \in V(O)$ and $V(d)(f) = x$, $V(c)(f) = y$ and define the context $\Gamma: \mathcal{L}_{\text{rg}} \rightarrow \mathbf{Set}$ by

$$\Gamma(I) = \emptyset, \Gamma(A) = \{f\}, \Gamma(O) = \{x, y\}$$

and the obvious action of the maps on f . Then we can think of Γ as the context

$$\{x: O, y: O, f: A(x, y)\}$$

which we will abbreviate to $\{x, y, f\}$.

Example 1.10. An important example of a context morphism is given by the “contraction” morphism

$$\{x: K, y: K, p: x = K^1 y\} \rightarrow \{x: K, q: x =_K^1 x\}$$

that sends both x and y to x .

We can now define the set of \mathcal{L} -formulas and their associated contexts of free variables by simultaneous induction.

Definition 1.11 (Formulas and Sequents). \top and \perp are atomic formulas and $\text{FV}(\top) = \text{FV}(\perp) = \emptyset$. If ϕ, ψ are formulas then so are $\phi \wedge \psi$, $\phi \vee \psi$ and $\phi \rightarrow \psi$ with

$$\text{FV}(\phi \wedge \psi) = \text{FV}(\phi \vee \psi) = \text{FV}(\phi \rightarrow \psi) = \text{FV}(\phi) \cup \text{FV}(\psi)$$

Negation $\neg\phi$ can be defined as $\phi \rightarrow \perp$. For the quantifiers, assume that ϕ is a formula and that x is a variable of sort K in $\text{FV}(\phi)$. Then $\forall x: K. \phi$ and $\exists x: K. \phi$ are formulas provided that $x \in \text{FV}(\phi)^\uparrow$. In that case we have

$$\text{FV}(\forall x: K. \phi) = \text{FV}(\exists x: K. \phi) = \text{FV}(\phi) \cup \text{dep}(x) \setminus \{x\}$$

We define a *sequent* as a syntactic entity of the form

$$\Gamma \mid \phi \vdash \psi$$

where ϕ, ψ are \mathcal{L} -formulas and $\Gamma \supset \text{FV}(\phi) \cup \text{FV}(\psi)$. (We follow [Jac99] in using the notation $\Gamma \mid \phi \vdash \psi$ in order to avoid overloading subscripts.)

Remark 1.12. The justification for not having top-level sorts correspond to atomic formulas in the form of relation symbols is that it simplifies proofs about the whole syntax by allowing us to consider fewer cases. From now on we will allow ourselves to switch to the usual “sugared” form whenever convenient. In \mathcal{L}_{rg} , for example, $I(f, x)$ will be syntactic sugar for $\exists \tau: I(f, x). \top$.

Remark 1.13. For reasons that will become clear in Section 5, we can without loss of generality restrict ourselves to “singular” sequents, i.e. those that consist of single formulas on either side of the turnstile. For added generality, most of our proofs below will be carried out in terms of sequents rather than individual formulas, although this changes nothing of essence in our arguments.

It remains to define substitution. Let Γ, Δ be (well-formed) contexts and $s: \Gamma \Rightarrow \Delta$ a context morphism. We need to describe how such a context morphism acts on formulas. Let ϕ be a formula in context Γ . Then we define $s(\phi)$, the substitution of ϕ along s , as follows:

- If $\phi \equiv \top, \perp$ then $s(\phi) \equiv \top, \perp$
- If $\phi \equiv \psi \wedge \chi, \psi \vee \chi, \psi \rightarrow \chi$ then $s(\phi) \equiv s(\psi) \wedge s(\chi), s(\psi) \vee s(\chi), s(\psi) \rightarrow s(\chi)$
- If $\phi \equiv \exists x: K.\psi, \forall x: K.\psi$ then $s(\phi) \equiv \exists y: K.s(\psi), \forall y: K.s(\psi)$ where y is fresh for $\Gamma \cup \Delta \cup \{x\}$

Clearly, in the last clause, there are many distinct choices of y . As Makkai also notes, this makes $s(\phi)$ not a well-defined operation. There are many ways of rectifying this, e.g. by imposing a well-ordering on variables or by defining the action of s on α -equivalence classes of formulas rather than formulas themselves. Since in this paper we are primarily concerned with describing n -logic at a high level of generality, we will simply assume that we have chosen one such way of making substitution well-defined on formulas. (Needless to say, different choices of achieving this may be available to us depending on whether we are using set theory or HoTT as our metatheory.)

2. SYNTAX OF n -LOGIC

We will now introduce the syntax of n -logic as an extension of the syntax of FOLDS described in Section 1. The key construction here is an operation that “adds equalities” to FOLDS signatures. This operation is suggestively called globular completion. After this is done, we are allowed to “extend” globularly completed signatures by new sorts that may depend on the equality sorts we just added. The signatures of n -logic are then obtained as arbitrarily large finite iterations of this process of first globularly completing and then extending. Given these signatures, the syntax is then defined as in Section 1. The syntax of n -logic is thus a formalization of the idea that we “add equalities” (globular completion) and then by extending add predicates and relations that talk *about* these equalities.

Definition 2.1 (Globular Completion). Let (\mathcal{L}, d) be a FOLDS signature of height n . The *globular completion* $(\mathcal{L}^=, d^=)$ of (\mathcal{L}, d) is given by the following data:

- (1) $\mathcal{L}^=$ contains all of \mathcal{L}
- (2) For each $K \in \mathcal{L}$ with $d(K) > -1$, $\mathcal{L}^=$ contains kinds $=_K^1, \dots, =_K^{d(K)+1}$ and arrows

$$s_i^K, t_i^K : =_K^i \rightarrow =_K^{i-1}$$

that satisfy the globular identities

$$\begin{aligned} s_{i-1}^K \circ s_i^K &= s_{i-1}^K \circ t_i^K \\ t_{i-1}^K \circ s_i^K &= t_{i-1}^K \circ t_i^K \end{aligned}$$

- (3) For any $K \in \mathcal{L}$ with $0 \leq d(K) < n$ and any $f: K \rightarrow K'$ we add the relation $f \circ s_1 = f \circ t_1$.
- (4) For each $=_K^j$ with $j \leq n$ we add a sort r_K^j and an arrow

$$\rho_j^K : r_K^j \rightarrow =_K^j$$

such that

$$s_j^K \circ \rho_j^K = t_j^K \circ \rho_j^K$$

(5) We define $d^=$ as follows:

- $d^=(K) = d(K)$ for all $K \in \text{ob}\mathcal{L}$
- $d^=(\text{=}_K^i) = d(K) - i$
- $d(r_K^i) = d(K) - (i + 1)$

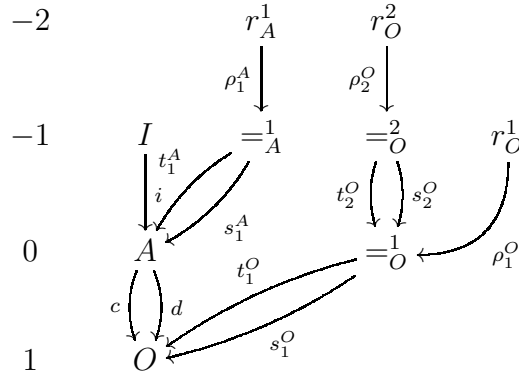
$(\mathcal{L}^=, d^=)$ is thus a FOLDS signature of height n (with extra structure in the form of specified sorts). We will call these new sorts *logical sorts*.

Remark 2.2. Clearly, s and t are to be understood as “source” and “target” maps for equality “paths”. On the other hand, r is to be understood as the predicate picking out the “reflexivity” proof of equality and the identities are there to ensure that r only applies to those equality sorts declared with identical variables. We will usually suppress explicit mention of K in the s , t and r .

Remark 2.3. The reflexivity sorts of dimension -2 play essentially no role in the syntax and we will usually omit writing them out in our examples. They are there to ensure the reflexivity of equality sorts of dimension -1 . The reason we add them is that they allow a more uniform presentation of the proof system in Section 5 in that all the properties of equality can be expressed by single set of rules.

Remark 2.4. We will also stipulate that the globular completion operation is idempotent. Once it has already been applied to (\mathcal{L}, d) , reapplying it does nothing.

Example 2.5. Consider \mathcal{L}_{rg} . Then $\mathcal{L}_{\text{rg}}^=$ is the following signature

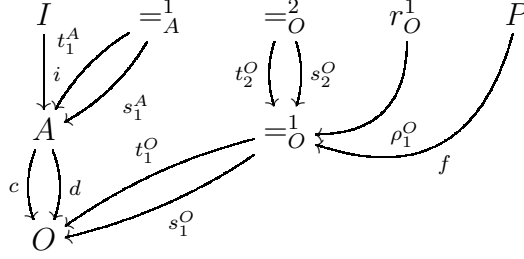


subject to the following extra relations:

$$\begin{aligned}
 s_1^O \circ s_2^O &= s_1^O \circ t_2^O, t_1^O \circ s_2^O = t_1^O \circ t_2^O \\
 t_1^O \circ \rho_1^O &= s_1^O \circ \rho_1^O, c \circ t_1^A c = c \circ s_1^A c \\
 t_2^O \circ \rho_2^O &= s_2^O \circ \rho_2^O, t_1^A \circ \rho_1^A = s_1^A \circ \rho_1^A \\
 d \circ t_1^A c &= d \circ s_1^A c
 \end{aligned}$$

Crucially, extensions of globularly completed signatures allow us to express properties and impose structure on the identity sorts themselves. For example, the following is

an extension of \mathcal{L}_{rg}



We will put this idea to use in axiomatizing univalent categories (Proposition 7.3).

Remark 2.6. Note that isomorphic categories with different dimension functions will generally give rise to very different globular completions, even if they are of the same height. For example, the signature $(\mathcal{L}_{\text{graph}}, d)$ with $d(O) = 2$ and $d(A) = 0$ will give a different globular completion from $(\mathcal{L}_{\text{graph}}, d')$ with $d'(O) = 2$ and $d'(A) = 1$. Syntactically, one can think of the dimension of each sort as specifying how “tall” its globular completion will be. Semantically, as we shall see, the dimension determines the given sort’s h -level.

Let us write Λ_n^0 for the set of FOLDS signatures of height n . We define

$$\Lambda_n^1 = \{(\mathcal{L}^=, d^=) | (\mathcal{L}, d) \in \Lambda_n^0\}$$

and

$$\overline{\Lambda}_n^1 = \{(\mathcal{L}', d') | (\mathcal{L}', d') > (\mathcal{L}, d) \in \Lambda_n^1\}$$

where $>$ is the relation of extension in Definition 1.6. We can now write

$$\Lambda_n^{i+1} = \{(\mathcal{L}^=, d^=) | (\mathcal{L}, d) \in \overline{\Lambda}_n^i\}$$

for arbitrary $0 \leq i \in \mathbb{N}$ and then define

$$\Lambda_n = \bigcup_{i \in \mathbb{N}} \Lambda_n^i$$

Λ_n is the set of signatures for n -logic, or n -signatures. These signatures are FOLDS signatures in the sense of Section 1, except they also contain extra structure in the form of the logical sorts $=^i_K, r^i_K$. We then define the *syntax of n -logic* simply to be the FOLDS syntax for signatures in Λ_n . Thus, given an n -signature \mathcal{L} , the Λ_n -formulas (resp. sentences, sequents etc.) for \mathcal{L} are simply the FOLDS \mathcal{L} -formulas (resp. sentences, sequents etc.) as described in Section 1.

Remark 2.7. It is straightforward to see that if $(\mathcal{L}, d) \in \Lambda_n$ then $(\mathcal{L}, d) \in \Lambda_n^i$ for some $i \leq n + 1$ and so Λ_n can be obtained only after a finite number of steps. (We could have avoided the redundancy, for example, by defining extensions as only those signatures that add sorts dependent on previously-introduced equality sorts.)

Remark 2.8. Λ_n can straightforwardly be described as the (category of) algebras of a monad on the category of FOLDS signatures (with possibly specified logical sorts) and morphisms the dimension-preserving extensions that preserve these logical sorts on the nose. In future work the “monadic packaging” of the syntax will likely prove

necessary if more general results are to be proved for n -logic involving higher n (e.g. a general completeness theorem).

3. HOMOTOPY SEMANTICS OF n -LOGIC

We will write out the semantics for n -logic in the syntax of (intensional) MLTT with homotopy type theory as our metatheory. (HoTT is here understood as intensional MLTT with at least one univalent universe and some higher inductive types). More generally, our semantics could be defined in any dependent type theory with a good notion of h -levels, or indeed in any categorical model of such a type theory (e.g. contextual categories [Car86] or C-systems [Voe14]).

Notation. We will write \mathcal{U} for a univalent universe of types and $\|A\|$ for the propositional truncation of a type A in \mathcal{U} . For any type A we will write Id_A^i for the i^{th} -iterated identity type. Thus, for example, $\text{Id}_A^3(\alpha, \beta)(p, q)(x, y)$ stands for

$$\text{Id}_{\text{Id}_{\text{Id}(x, y)}(p, q)}(\alpha, \beta)$$

Otherwise we will follow the notation of [Uni13] closely.

Remark 3.1. It is unclear how weak the metatheory can be made if we are to be able to prove completeness as we do in the sequel. For example, the use of higher inductive types in the proof of completeness seems to us necessary, so it is reasonable to expect that a system that cannot account for the higher inductive types defined there will not be able to serve as our metatheory. For example, though the semantics presented here can certainly be defined in Cubical Type Theory [CCHM15] we are not yet certain that the completeness theorem can be proved if we take it as our metatheory.

Fix $n \geq -1$ and $(\mathcal{L}, d) \in \Lambda_n$. We write K_i^j for each object in \mathcal{L} where $j = l(K_i^j)$. Similarly, we write f_{ijkl} for each morphism $K_i^j \rightarrow K_k^l$ in \mathcal{L} . For each K_i^j in \mathcal{L} we write $a_{ijkl} \in \mathbb{N}$ for the cardinality of the set of all non-identity morphisms from $K_i^j \rightarrow K_k^l$, i.e. $a_{ijkl} = |\mathcal{L}(K_i^j, K_k^l)|$. We write a_{ij} for the cardinality of the set of all non-identity morphisms out of K_i^j in \mathcal{L} and A_{ij} for that set of morphisms, i.e. $a_{ij} = |A_{ij}|$. It will also be convenient to write D_{ij} for the set of lower indices of the codomains of each morphism in A_{ij} , i.e. $D_{ij} = \{k | f_{ijkl} \in A_{ij}\}$. We will also assume there is an ordering on A_{ij} such that we have an induced order-preserving indexing $p^{ij}: \{1, \dots, a_{ij}\} \rightarrow A_{ij}$. Thus when we write p_k^{ij} this means the k -th morphism out of K_i^j for the given ordering. Similarly, we will assume that there is an indexing of the codomains of these morphisms $d^{ij}: \{1, \dots, a_{ij}\} \rightarrow D_{ij}$ such that d_k^{ij} is the lower index of the codomain of p_k^{ij} .

The basic idea of an \mathcal{L} -structure is as follows. First we give the denotation of the non-logical sorts by induction on their level: “bottom-level” sorts K are n -types and sorts of level $m \geq n$ are dependent functions into $d(K)$ -types which rely on the definitions of the non-logical sorts of lower level. Logical sorts are then defined as the identity sorts and reflexivity predicates on sorts that have already received denotation. We then assign denotations to sorts that may depend on logical sorts.

We repeat the process until everything has been assigned a denotation. The following (long) definition spells out this process.

Definition 3.2 (Homotopy \mathcal{L} -structure). An \mathcal{L} -structure \mathcal{M} (or \mathcal{L} - n -structure if we want to make n explicit) is obtained by the following process. We begin with non-logical sorts none of which depend on logical sorts (i.e. those that belong to the initial FOLDS signature, before it is globularly completed). We assign them denotations by induction on their level as follows:

$l=0$ For each K_i^0 we pick an n -type $\mathcal{M}(K_i^0)$: n -**type** $_{\mathcal{U}}$

$l=1$ For each K_i^1 we pick a term

$$\mathcal{M}(K_i^1): \mathcal{M}(K_{p_1^{i1}}^0) \rightarrow \mathcal{M}(K_{p_2^{i1}}^0) \rightarrow \dots \rightarrow \mathcal{M}(K_{p_{a_{i1}}^{i1}}^0) \rightarrow d(K_i^1)\text{-}\mathbf{type}_{\mathcal{U}}$$

$l=2$ For each K_i^2 we pick a term

$$\begin{aligned} \mathcal{M}(K_i^2): \prod_x \mathcal{M}(K_{d_1^{i2}}^1)(x_{p_1^{i2_1} \circ p_1^{i2}}, \dots, x_{p_{a_{i2}}^{i2_1} \circ p_1^{i2}}) &\rightarrow \mathcal{M}(K_{d_2^{i2}}^1)(x_{p_1^{i2_2} \circ p_2^{i2}}, \dots, x_{p_{a_{i2}}^{i2_2} \circ p_2^{i2}}) \\ &\rightarrow \dots \rightarrow \mathcal{M}(K_{p_{a_{i2}}^1}^1) \rightarrow d(K_i^2)\text{-}\mathbf{type}_{\mathcal{U}} \end{aligned}$$

Beyond $l = 2$ it becomes very difficult to write down the relevant types while keeping all variable dependencies explicit. Suppressing variable dependencies we can thus write the data for a sort of level n (equivalently, of dimension -1) as follows:

$$\mathcal{M}(K_i^n): \underbrace{\prod_{\vec{x}_0} \prod_{\vec{x}_1} \dots \prod_{\vec{x}_{n-2}}}_{x_k^i: K_j^i \text{ for some } j} \mathcal{M}(K_{d_1^{in}}^{n-1}) \rightarrow \dots \rightarrow \mathcal{M}(K_{d_{a_{in}}^{in}}^{n-1}) \rightarrow (-1)\text{-}\mathbf{type}_{\mathcal{U}}$$

On the other hand, for logical sorts we do the following, writing $K^{\mathcal{M}}$ for the denotation of K in \mathcal{M} and $\Gamma_{K^{\mathcal{M}}}$ for its canonical context. Then for any non-logical $K \in \text{ob}\mathcal{L}$ we let $(=^i_K)^{\mathcal{M}}$ be the i^{th} -iterated identity type “over” $K^{\mathcal{M}}$:

$$\begin{aligned} (=^i_K)^{\mathcal{M}} &\equiv \lambda \vec{x}: \Gamma_{K^{\mathcal{M}}} . \lambda y, z: K^{\mathcal{M}}(\vec{x}). \\ &\quad \lambda p_1, q_1: \text{Id}_{K^{\mathcal{M}}}(\vec{x})(y, z) \dots \lambda p_i, q_i: \text{Id}_{K^{\mathcal{M}}}^i(\vec{x})(p_{i-1}, q_{i-1}) \dots (p_1, q_1) \\ &\quad : \prod_{\substack{\vec{x}: \Gamma_{K^{\mathcal{M}}} \\ y, z: K^{\mathcal{M}}(\vec{x})}} (d(K) - i)\text{-}\mathbf{type}_{\mathcal{U}} \\ p_j, q_j: &\text{Id}_{K^{\mathcal{M}}}^j(\vec{x})(y, z)(p_1, q_1) \dots (p_{j-1}, q_{j-1}) \\ &1 \leq j \leq i-1 \end{aligned}$$

For reflexivity sorts we define

$$\begin{aligned} r_K^i &\equiv \lambda \vec{x}: \Gamma_{K^{\mathcal{M}}} . \lambda y, z: K^{\mathcal{M}}(\vec{x}) \dots \lambda q: \text{Id}_{K^{\mathcal{M}}}^{i-1}(y, z) \dots (p_{i-2}, q_{i-2}). \\ \lambda p: &\text{Id}_{K^{\mathcal{M}}}^i(y, z) \dots (q, q) . || \text{Id}(p, \text{refl}_q) || : \prod_{\substack{\vec{x}: \Gamma_{K^{\mathcal{M}}} \\ y, z: K^{\mathcal{M}}(\vec{x})}} (d(K) - (i+1))\text{-}\mathbf{type}_{\mathcal{U}} \\ p_j, q_j: &\text{Id}_{K^{\mathcal{M}}}^j(\vec{x})(y, z)(p_1, q_1) \dots (p_{j-1}, q_{j-1}) \\ &1 \leq j \leq i-2 \\ q: &\text{Id}_{K^{\mathcal{M}}}^{i-1}(y, z) \dots (p_{i-2}, q_{i-2}) \\ p: &\text{Id}_{K^{\mathcal{M}}}^{i-1}(y, z) \dots (q, q) \end{aligned}$$

Thus all “first-generation” sorts and their identity sorts and reflexivity relations have been assigned denotation. We now repeat the same process for later generations

until everything has been assigned a denotation. This completes the definition of an \mathcal{L} -structure \mathcal{M} .

Remark 3.3. Clearly, the definition of an \mathcal{L} -structure is not functorial in the sense that the relations in \mathcal{L} encoding variable dependencies are “written in” as judgmental equalities by hand, rather than us first coding up \mathcal{L} as a category in HoTT and defining an \mathcal{L} -structure as a functor out of it. (Another way of saying this is that our definition of an \mathcal{L} -structure is given in the “Reedy way”.) This is a well-known limitation in intensional MLTT that hinders the uniformity of our definition. We console ourselves in the fact that given a finite signature \mathcal{L} it will always in principle be possible to carry out the process described in Definition 3.2. Doing so efficiently (even algorithmically) is another matter altogether.

Remark 3.4. The fact that $(=^i_K)^{\mathcal{M}}$ is indeed a term of the above-specified type depends on us having asserted that each $K^{\mathcal{M}}(\vec{x})$ will be a $d(K)$ -type (i.e. that we have a proof of that fact in hand) and similarly for the denotation of reflexivity sorts.

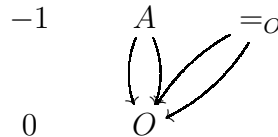
Remark 3.5. An \mathcal{L} - n -structure \mathcal{M} may depend on some (non-empty) ambient context (i.e. be an “interpretation with parameters”). In that case we will denote such an ambient context by $\Delta^{\mathcal{M}}$. For example, in an \mathcal{L} - n -structure-with-parameters \mathcal{M} , the type declarations of ground sorts K are to be understood as claiming that the following judgment is derivable

$$\Delta^{\mathcal{M}} \vdash \mathcal{M}(K) : n\text{-type}_{\mathcal{U}}$$

where, as before, we abuse notation in identifying the pair K with its first projection.

Example 3.6. Let $\mathcal{L} = \mathbf{1}$, where $\mathbf{1}$ is the category with one object and one identity arrow. Then an $\mathbf{1}$ - (-1) -structure is simply a mere proposition $P : \mathbf{Prop}_{\mathcal{U}}$. This allows us to say that (-1) -logic has the same expressive power as propositional logic. Similarly, a $\mathbf{1}$ - n -structure is simply an n -type. Thus the empty $\mathbf{1}$ - n -theory is simply the theory of n -groupoids (understood as n -types in HoTT). This provides a generalization of the fact that the empty theory over a signature with a single sort in first-order logic is simply the theory of sets, also known as the “pure theory of identity”.

Example 3.7. An $\mathcal{L}_{\text{graph}}$ -structure consists of an h -set $O : \mathcal{U}$ and a mere relation $A : O \rightarrow O \rightarrow \mathbf{Prop}_{\mathcal{U}}$. More generally, the study of 0-signatures coincides with traditional set-based model theory (with the restriction that we are only considering relation symbols). 0-logic can thus be thought of as the “classical limit” of n -logic. Consider $\mathcal{L}_{\text{graph}}^=$ as an illustration:



$\mathcal{L}_{\text{graph}}$ -0-formulas are then exactly (when suitably translated) the formulas of first-order logic with equality for a single-sorted signature Σ with a single binary predicate A . Semantically, an $\mathcal{L}_{\text{graph}}$ -0-structure \mathcal{M} consists of a 0-type $O^{\mathcal{M}} : \mathbf{Set}_{\mathcal{U}}$ and a dependent type $A^{\mathcal{M}} : O^{\mathcal{M}} \rightarrow O^{\mathcal{M}} \rightarrow \mathbf{Prop}_{\mathcal{U}}$. This is all entirely analogous to Σ -structures

in traditional set-theoretic semantics. We may say that 0-logic has the same expressive power as first-order logic with equality and its semantics as defined here coincide with the usual set-theoretic semantics.

Example 3.8. An $\mathcal{L}_{\text{rg}}\text{-1}$ -structure \mathcal{M} consists of the following data:

$$\begin{aligned} I^{\mathcal{M}} &: \prod_{x: O^{\mathcal{M}}} A(x, x) \rightarrow \mathbf{Prop}_{\mathcal{U}} \\ A^{\mathcal{M}} &: O^{\mathcal{M}} \rightarrow O^{\mathcal{M}} \rightarrow \mathbf{Set}_{\mathcal{U}} \\ O^{\mathcal{M}} &: \mathbf{Gpd}_{\mathcal{U}} \end{aligned}$$

Given a context Γ (in \mathcal{L}) written in the MLTT style as

$$\Gamma = \{x_1: K_1, x_2: K_2(x_1), \dots, x_n: K_n(x_1, \dots, x_{n-1})\}$$

we obtain a well-formed context in MLTT as follows

$$\Gamma^{\mathcal{M}} = \{x_1: K_1^{\mathcal{M}}, \dots, x_n: K_n^{\mathcal{M}}(x_1, \dots, x_{n-1})\}$$

where we keep the variable names the same for convenience. (When Γ is empty we obviously take $\Gamma^{\mathcal{M}}$ also to be the empty context.) A context morphism (in MLTT) $\vec{a}: \Delta^{\mathcal{M}} \rightarrow \Gamma^{\mathcal{M}}$ is then called a *realization* of Γ in \mathcal{M} .

With this in mind, we now define the interpretation of \mathcal{L} -formulas in an \mathcal{L} - n -structure \mathcal{M} . This proceeds much as one would expect: universal quantifiers become Π -types and existential quantifiers become Σ -types and conjunction, disjunction, implication and negation are translated in the usual manner, namely as the type formers \times , $+$, \rightarrow and $(-) \rightarrow \mathbf{0}$ respectively.

Definition 3.9 (Interpretation of Formulas). Let ϕ and ψ be \mathcal{L} -formulas. We interpret them as types in HoTT as follows:

$$\begin{aligned} \top^{\mathcal{M}} &=_{\text{def}} \mathbf{1} \\ \perp^{\mathcal{M}} &=_{\text{def}} \mathbf{0} \\ (\phi \wedge \psi)^{\mathcal{M}} &=_{\text{def}} \phi^{\mathcal{M}} \times \psi^{\mathcal{M}} \\ (\phi \vee \psi)^{\mathcal{M}} &=_{\text{def}} \|\phi^{\mathcal{M}} + \psi^{\mathcal{M}}\| \\ (\phi \rightarrow \psi)^{\mathcal{M}} &=_{\text{def}} \phi^{\mathcal{M}} \rightarrow \psi^{\mathcal{M}} \\ (\neg \phi)^{\mathcal{M}} &=_{\text{def}} \phi^{\mathcal{M}} \rightarrow \mathbf{0} \\ (\exists x: K.\phi(x))^{\mathcal{M}} &=_{\text{def}} \|\sum_{x: K^{\mathcal{M}}} \phi^{\mathcal{M}}\| \\ (\forall x: K.\phi(x))^{\mathcal{M}} &=_{\text{def}} \prod_{x: K^{\mathcal{M}}} \phi^{\mathcal{M}} \end{aligned}$$

Remark 3.10. There is a certain ambiguity in Definition 3.9 since what is on the right hand side is meant to be interpreted as a type, yet it is possible that there are free variables floating around. Rather, the above definitions should be understood as saying that formulas are interpreted as judgements that the right-hand side is a type in some context that contains its free variables.

Example 3.11. In \mathcal{L}_{rg} if we are given the formula $\phi \equiv \exists \tau: I(x, f). \top$ then its interpretation in some \mathcal{L}_{rg} -structure \mathcal{M} will be given by $\|\sum_{\tau: I^{\mathcal{M}}(x, f)} \mathbf{1}\|$ which is of course equivalent to $\|I^{\mathcal{M}}(x, f)\|$. But this is not strictly speaking a type since x and f remain variables. Instead, we should take the interpretation of ϕ in \mathcal{M} to consist of the following judgement

$$\Delta^{\mathcal{M}}, x: O^{\mathcal{M}}, f: A^{\mathcal{M}}(x, x) \vdash \|I^{\mathcal{M}}(x, f)\|: \mathcal{U}$$

where we are abusing notation in using $I^{\mathcal{M}}(x, f)$ for what really is its first projection since strictly speaking $I^{\mathcal{M}}$ was defined as a dependent function into $\mathbf{Prop}_{\mathcal{U}}$. Since both $\mathbf{0}$ and $\mathbf{1}$ are types in the empty context, by the weakening rule for contexts we will get an analogous definition for any formula ϕ in any context, possibly larger than its context of free variables.

Remark 3.12. As the above example illustrates, all non-trivial propositions are to be constructed using the existential quantifier. As is reasonable, we will refrain from interpreting a formula $\exists x: A. \top$ as $\|\sum_{x: A^{\mathcal{M}}} \mathbf{1}\|$ and write it instead as $\|A^{\mathcal{M}}\|$. The latter two types are of course equivalent.

We are now ready to define a notion of satisfaction for formulas and sequents.

Definition 3.13 (Satisfaction). Let ϕ be any \mathcal{L} - n -formula, \mathcal{M} any \mathcal{L} - n -structure, \vec{a} any realization of a context $\Gamma \supset \text{FV}(\phi)$ in \mathcal{M} . We define *satisfaction of ϕ by \vec{a} in \mathcal{M}* as follows:

$$\mathcal{M} \models \phi[\vec{a}/\Gamma] \text{ iff } \phi^{\mathcal{M}}[\vec{a}/\Gamma] \text{ is inhabited}$$

The case where ϕ has no free variables is a special case of the above definition, in which case we write $\mathcal{M} \models \phi$ and say that \mathcal{M} is a model of ϕ . Satisfaction for sequents can be defined similarly and is suggested by their very notation: For sequents we define satisfaction as follows:

$$\mathcal{M} \models \Gamma \mid \phi \vdash \psi \text{ iff } \Delta^{\mathcal{M}}, \Gamma^{\mathcal{M}}, x: \phi^{\mathcal{M}} \vdash y: \psi^{\mathcal{M}} \text{ is derivable}$$

Remark 3.14. The right-hand side of the first biconditional in Definition 3.13 is equivalent to requiring that a judgement

$$\Delta^{\mathcal{M}} \vdash \pi: \phi^{\mathcal{M}}[\vec{a}/\Gamma]$$

is derivable in HoTT, where π is a (metavariable for a) term of the relevant type.

Remark 3.15. Since we will always assume the presence of Π -types in our metatheory, the right-hand side of the second biconditional in Definition 3.13 is equivalent to the statement that the type

$$\prod_{\vec{x}: \Gamma^{\mathcal{M}}} \phi^{\mathcal{M}} \rightarrow \psi^{\mathcal{M}}$$

is inhabited (in context $\Delta^{\mathcal{M}}$). We will most often interpret sequents in this latter form, since it is the more convenient (albeit less general) of the two.

Example 3.16. Consider the following \mathcal{L}_{rg} -sentence

$$\phi \equiv \forall x: O \exists f: A(x, x). I(f)$$

and let \mathcal{M} be the \mathcal{L}_{rg} -1-structure given by the following data

$$\langle \mathbf{Set}_{\mathcal{U}}, \lambda x. \lambda y. x \rightarrow y, \lambda x. \lambda f. \text{Id}_{x \rightarrow x}(f, 1_x) \rangle$$

with ambient context \emptyset . Then \mathcal{M} is a model of ϕ . To see this, observe that

$$\mathcal{M} \models \forall x: O \exists f: A(x, x). I(f)$$

since

$$\emptyset \vdash \lambda x. (1_x, \text{refl}_{1_x}): \prod_{x: \mathbf{Set}_{\mathcal{U}}} \parallel \sum_{f: x \rightarrow x} \text{Id}_{x \rightarrow x}(f, 1_x) \parallel$$

is derivable in UF and since

$$\forall x: O \exists f: A(x, x). I(f))^{\mathcal{M}} \equiv \prod_{x: \mathbf{Set}_{\mathcal{U}}} \parallel \sum_{f: x \rightarrow x} \text{Id}_{x \rightarrow x}(f, 1_x) \parallel$$

Finally, note that the way we have set up our semantics all the types that interpret sentences and substitution instances of formulas will be h -props (“mere propositions”). This is because every atomic formula is a mere proposition and all logical constructors preserve h -props except $+$ and Σ . But for the latter we have taken their propositional truncation, as is also done in [Uni13]. We can record this fact as a proposition, since the soundness of our proof system \mathcal{D} introduced in the next Section crucially depends on it.

Proposition 3.17. *For any \mathcal{L} -formula ϕ , \mathcal{L} - n -structure \mathcal{M} , context $\Gamma \supset FV(\phi)$ and realization \vec{a} of Γ in \mathcal{M} we have that $\phi^{\mathcal{M}}[\vec{a}/\Gamma]$ is a mere proposition.*

4. SET-THEORETIC SEMANTICS FOR 1-LOGIC

The semantics we have presented in Section 3 use HoTT as a metatheory and thus take place within Univalent Foundations. In this section we give a semantics for n -logic in set theory. The easiest way of doing this is would be simply to interpret the syntax of n -logic in a category with the appropriate structure to interpret the required type constructors (e.g. C -systems or contextual categories) or even directly into the category of Kan complexes in simplicial sets, following the “canonical” model of homotopy type theory constructed in [KL14]. However, such a semantics does not add anything essentially new to the homotopy semantics presented in Section 3. A set-theoretic semantics for n -logic becomes interesting only when we have in our possession an independent (and independently interesting) description of homotopy n -types. Such is certainly the case for $n = 1$ where groupoids admit a very simple description as categories all of whose morphisms are invertible. We will now give such a semantics by extending Makkai’s original functorial semantics for FOLDS. Throughout, we will assume that we are working in a classical set theory, i.e. one in which subobject lattices are Boolean.

As before, fix a 1-signature \mathcal{L} . Let V be a (strict, i.e. with on-the-nose preservation of identities and compositions) functor as in Definition 1.7 except we now postcompose it with the inclusion $\mathbf{Set} \hookrightarrow \mathbf{Gpd}$ so that view it as a functor

$$V: \mathcal{L} \rightarrow \mathbf{Gpd}$$

where **Gpd** is the 2-category of groupoids viewed as a full subcategory of the 2-category of categories **Cat**. As before, a context is a finite (necessarily strict) subfunctor Γ of V and a context morphism a (necessarily strict) natural transformation between such subfunctors.

Definition 4.1 (Set-theoretic \mathcal{L} -structure). A (*set-theoretic*) \mathcal{L} -1-structure \mathcal{M} is a pseudofunctor $(\mathcal{M}, \mu): \mathcal{L} \rightarrow \mathbf{Gpd}$ such that:

- (1) For each K with $d(K) = 1$ we fix

$$\mathcal{M}(=^1_K) \equiv \text{mor}\mathcal{M}(K)$$

where we regard $\text{mor}\mathcal{M}(K)$ as the arrow groupoid $\mathcal{M}(K)^{\bullet \rightarrow \bullet}$ and interpret $\mathcal{M}(s^1_K)$ and $\mathcal{M}(t^1_K)$ as the domain and codomain functors (indeed, fibrations)

- (2) For each K with $d(K) = 0$ we interpret $=^1_K$ as equality of objects of the groupoid $\mathcal{M}(K)$, i.e. fix

$$\mathcal{M}(=^1_K) \equiv \{(a, a) | a \in \text{ob}\mathcal{M}(K)\}$$

regarded as a discrete groupoid and with $\mathcal{M}(s^1_K)$ and $\mathcal{M}(t^1_K)$ interpreted as the obvious projections.

- (3) We fix

$$\mathcal{M}(r^1_K) \equiv \{1_a | a \in \text{ob}\mathcal{M}(K)\}$$

regarded as a discrete groupoid and with $\mathcal{M}(\rho^1_K)$ the obvious inclusion.

Remark 4.2. Clearly, not every functor $F: \mathcal{L} \rightarrow \mathbf{Gpd}$ is an \mathcal{L} -structure since we require fixed denotations for logical sorts. Indeed, the variable functor V is not an \mathcal{L} -structure anymore as in Makkai's original definition, nor is any context Γ . Thus, we may no longer view both the syntax and the semantics of n -logic as constructed functorially in exactly the same way. This asymmetry between syntax and semantics is inevitable when we move to n -logic for $n \geq 1$. This is because we want variables (as syntactic entities) to have strict identity conditions, i.e. we want them to be equal or unequal (and decidable so). Yet in n -logic such variables may denote entities with much coarser identity conditions, e.g. objects in a groupoid. Another way to think about this point is that if p is variable of sort $x =^1_K y$ (i.e. $p \in V(= K^1)$) and $V(s^1_K)(p) = x$ and $V(t^1_K)(p) = y$ then p does not denote an actual isomorphism between x and y because x and y are assumed distinct and so there should be no isomorphism (or equality) between them in $V(K)$. However, in proving completeness for the set-theoretic semantics of 1-logic in the sequel we will consider a variable functor V that is actually an \mathcal{L} -structure, i.e. for which (certain) sorts are interpreted as (proper) groupoids. To do this we construct a “generic” *groupoid* of variables in much the same way that \mathbb{N} can be seen as a “generic” *set* of variables.

Recall that a *pseudonatural transformation* (α, η) between pseudofunctors

$$(F, \phi), (G, \psi): \mathcal{K}_1 \rightarrow \mathcal{K}_2$$

is a natural transformation in which naturality (i.e. commutation of the required squares) holds only up to coherent isomorphisms given by η . (For a precise definition

cf. B1.1.2 [Joh03].) We now want to consider evaluations as pseudonatural transformations from contexts to \mathcal{L} -structures. In our situation the definition is simplified significantly by the fact that the codomain of our (pseudo)functors will always be a 1-category and the codomain functor takes values in discrete categories.

Definition 4.3 (Evaluation). Given a context Γ and an \mathcal{L} -1-structure \mathcal{M} , an *evaluation of Γ in \mathcal{M}* is a pseudonatural transformation

$$(\alpha, \eta): \Gamma \Rightarrow (\mathcal{M}, \mu): \mathcal{L} \rightarrow \mathbf{Gpd}$$

Explicitly an evaluation is given by the following data:

- (1) A collection of maps $\{\alpha_K: \Gamma(K) \rightarrow \mathcal{M}(K)\}_{K \in \text{ob } \mathcal{L}}$
- (2) For every $f: K \rightarrow K_f$ in \mathcal{L} a natural isomorphism

$$\eta_f: \mathcal{M}(f) \circ \alpha_K \rightarrow \alpha_{K_f} \circ \Gamma(f)$$

(where \circ denotes horizontal composition of 2-cells) such that for any $g: K_f \rightarrow K_{gf}$ we have

$$\eta_{gf} = \eta_g \circ g(\eta_f) \circ \mu_{f,g}$$

We will write $\mathbf{pNat}(\Gamma, \mathcal{M})$ for the set of pseudonatural transformations between pseudofunctors Γ and \mathcal{M} .

Remark 4.4. The “pseudo” part for natural transformations is essential: without it our semantics will not be sound with respect to the proof system developed in Section 5. On the other hand, the “pseudo” can be dropped from the definition of \mathcal{L} -structures in the sense that completeness can be proved with respect to the class of strict \mathcal{L} -structures (i.e. those defined by strict functors $\mathcal{L} \rightarrow \mathbf{Gpd}$). Nevertheless, our semantics at this point can only gain from the added generality, and so we will not impose this restriction on ourselves.

Let us write y for the Yoneda embedding post-composed with the inclusion $\mathbf{Set} \hookrightarrow \mathbf{Gpd}$ and \hat{y} for the subfunctor of y that misses the identity on the given object, i.e. $\hat{y}X(X) = yX(X) \setminus \{1_X\}$. Given a variable $x \in |\Gamma|$ we can define $\partial x: \hat{y}K \Rightarrow \Gamma$ by $f \mapsto \Gamma(f)(x)$. Given any evaluation (α, η) and any variable $x: K$ we can define the α -boundary of x to be the composite $\alpha \circ \partial x$, and we write it as $\partial_\alpha x$. (Note that since ∂x is a strict natural transformation between strict functors, $\alpha \circ \partial x$ become a pseudonatural transformation simply by inheriting η .) We also write $\alpha * a$ for the class of maps (which is not necessarily a (pseudo)natural transformation) that takes x to a and restricts to α otherwise.

The key difference from standard semantics of FOLDS is in how we define the x -range of α . Intuitively we want this to be the set of all those terms in \mathcal{M} that could serve as interpretations of the variable x in the given evaluation α . For this to make sense we must require that these terms $a \in \text{ob } \mathcal{M}(K)$ are “supported” by coherent isomorphisms that can extend α into a pseudonatural transformation that takes x to a . This is the idea behind the following definition, which we can think of as the η -coherent x -range of α .

Definition 4.5 (*x*-range of an evaluation). Let (α, η) be an evaluation of Γ in \mathcal{M} and $x \in \Gamma^\uparrow$. Then the *x*-range of (α, η) is defined as the following set

$$\mathcal{M}_{(\alpha, \eta)}[x : K] = \{a \in \text{ob}\mathcal{M}(K), \bar{\eta} = (\bar{\eta}_f : \mathcal{M}(f)(a) \rightarrow (\partial_\alpha x)_{K_f})_{f : K \rightarrow K_f} \mid (\alpha * a, \eta * \bar{\eta}) \in \mathbf{pNat}(\{\Gamma, x : K\}, \mathcal{M})\}$$

where by $\eta * \bar{\eta}$ we mean the *amalgamation* of the two coherence data into a single family of maps. (One can think of $*$ simply as set-theoretic union if one takes this data to be given in the form of a set of arrows.) If $(a, \bar{\eta}) \in \mathcal{M}_\alpha[x]$, then we will write $\alpha[a/x]$ for the pseudonatural transformation that restricts to α in Γ and takes x to a , omitting the coherence data (as long as we know it exists, which might not always be the case).

Definition 4.6 (Extension of a formula). We define recursively the sets that comprise the *extension* $\mathcal{M}(\Gamma \mid \phi)$ of any \mathcal{L} -formula ϕ in context Γ interpreted in \mathcal{M} omitting explicit mention of coherence data:

$$\begin{aligned} \mathcal{M}(\Gamma \mid \top) &=_{\text{def}} \mathbf{pNat}(\Gamma, \mathcal{M}) \\ \mathcal{M}(\Gamma \mid \perp) &=_{\text{def}} \emptyset \\ \mathcal{M}(\Gamma \cup \Delta \mid \phi \wedge \psi) &=_{\text{def}} \mathcal{M}(\Gamma \mid \phi) \cap \mathcal{M}(\Delta \mid \psi) \\ \mathcal{M}(\Gamma \cup \Delta \mid \phi \vee \psi) &=_{\text{def}} \mathcal{M}(\Gamma \mid \phi) \cup \mathcal{M}(\Delta \mid \psi) \\ \mathcal{M}(\Gamma \cup \Delta \mid \phi \rightarrow \psi) &=_{\text{def}} \{\alpha \in \mathbf{pNat}(\Gamma, \mathcal{M}) \mid \text{If } \alpha \in \mathcal{M}(\Gamma \mid \phi) \text{ then } \alpha \in \mathcal{M}(\Gamma \mid \psi)\} \\ \mathcal{M}(\Gamma \mid \forall x \phi) &=_{\text{def}} \{\alpha \in \mathcal{M}(\Gamma \mid \phi) \mid \forall a \in \mathcal{M}_\alpha[x], \alpha[a/x] \in \mathcal{M}(\Gamma \cup \{x\} \mid \phi)\} \\ \mathcal{M}(\Gamma \mid \exists x \phi) &=_{\text{def}} \{\alpha \in \mathcal{M}(\Gamma \mid \phi) \mid \exists a \in \mathcal{M}_\alpha[x], \alpha[a/x] \in \mathcal{M}(\Gamma \cup \{x\} \mid \phi)\} \end{aligned}$$

For the quantifiers we have assumed that x is a variable of sort K free in ϕ and that $x \in \text{FV}(\phi)^\uparrow$.

Remark 4.7. The fact that we consider $\mathbf{pNat}(\Gamma, \mathcal{M})$ only as a set (ignoring extra structure that we may naturally put on it) is also essentially what makes our approach *proof-irrelevant*: we don't care about *how* the extension of a formula can be embedded in the extension of another formula, but merely *whether* it can.

Remark 4.8. Note that in the form presented in Definition 4.6 our semantics will always validate the LEM since we have defined it only in (a material and classical) set theory. In order to generalize the interpretation to constructive models (e.g. a Kripke-style semantics) we need to define it, as usual, for suitably-structured categories with non-Boolean subobject lattices. However, the equality sorts here make such a general definition more involved than usual. A big part of the sequel is devoted to defining this generalization, as a stepping stone to proving completeness.

Definition 4.9 (Satisfaction). For any \mathcal{L} -structure \mathcal{M} and ϕ any \mathcal{L} -formula we can define *satisfaction* as follows:

$$\mathcal{M} \models \phi[\alpha/\Gamma] \Leftrightarrow \alpha \in \mathcal{M}(\Gamma \mid \phi)$$

Similarly we define *satisfaction of sequents* as follows:

$$\mathcal{M} \models \Gamma \mid \phi \vdash \psi \Leftrightarrow \mathcal{M}(\Gamma \mid \phi) \subseteq \mathcal{M}(\Gamma \mid \psi)$$

Since an evaluation α already contains information about its domain Γ , we can simply write $\phi[\alpha]$ for the evaluation of ϕ at α . As one would expect, given any formula ϕ it always suffices to consider $\text{FV}(\phi)$ as its context. When $\text{FV}(\phi) = \emptyset$ we write $\mathcal{M} \models \phi$ if the unique pseudonatural transformation $! : \emptyset \Rightarrow \mathcal{M}$ is in $\mathcal{M}(\emptyset : \phi)$.

The usual substitution lemma ensuring that satisfaction is preserved under composition of evaluations (i.e. repeated substitutions) can be proved straightforwardly, noting the minor subtlety that we are composing pseudonatural transformations and not strict ones. We state it for the record, leaving the proof to the reader.

Lemma 4.10 (Substitution Lemma). *Let Γ, Δ be contexts, $\delta : \Delta \Rightarrow \Gamma$ a context morphism and ϕ a formula in context Δ . If $\alpha \in \mathcal{M}(\Gamma : \delta(\phi))$ then $\alpha \circ \delta \in \mathcal{M}(\Gamma : \phi)$. In other words:*

$$\mathcal{M} \models \delta(\phi)[\alpha] \Rightarrow \mathcal{M} \models \phi[\alpha \circ \delta]$$

Given any two pseudonatural transformations $(\alpha, \eta), (\beta, \theta)$ a *modification*

$$\nu : (\alpha, \eta) \rightarrow (\beta, \theta)$$

is given by maps $\nu_K : \alpha_K \rightarrow \beta_K$ satisfying evident coherence conditions with respect to η and θ . Since the elements of the extensions of formulas in our semantics are pseudonatural transformations, we can consider modifications between. In particular, we will say that two evaluations $\alpha, \beta \in \mathcal{M}(\Gamma : \phi)$ are *isomorphic*, and write $\alpha \cong \beta$, if there is a (necessarily invertible) modification $\nu : (\alpha, \eta) \rightarrow (\beta, \theta)$. Explicitly, in our simplified setting, such an isomorphism is given by the following data:

- (1) For each $K \in \text{ob}\mathcal{L}$ and $x \in \Gamma(K)$ an arrow $(\nu_K)_x : \alpha_K(x) \rightarrow \beta_K(x)$ in $\mathcal{M}(K)$ such that
- (2) for every $f : K \rightarrow K_f$ in \mathcal{L} we have

$$((\nu_{K_f})_{\Gamma(f)(x)}) \circ (\eta_f)_x = (\theta_f)_x \circ \mathcal{M}(f)((\nu_K)_x)$$

The following lemma establishes that our semantics does not distinguish between isomorphic evaluations.

Lemma 4.11 (“Invariance under modifications”). *Let $\mathcal{M} \models \phi[\alpha]$ and $\beta \cong \alpha$. Then $\mathcal{M} \models \phi[\beta]$.*

Proof. We proceed by induction on complexity. The cases of $\top, \perp, \vee, \wedge$ and \rightarrow follow immediately. We do the case of existential quantification. Let $(\alpha, \eta) \in \mathcal{M}(\Gamma \mid \exists x : K. \phi)$ and assume we are given $\nu : (\alpha, \eta) \rightarrow (\beta, \theta)$. This means there exist coherence data $(a, \bar{\eta}) \in \mathcal{M}_{(\alpha, \eta)}[x : K]$ such that $(\alpha[a/x], \eta * \bar{\eta}) \in \mathcal{M}(\Gamma, x : K \mid \phi)$. For any arrow $f : K \rightarrow K_f$ in \mathcal{L} define

$$\bar{\theta}_f = \nu_{K_f} \circ \bar{\eta}_f$$

It is easy to check that this defines coherence data such that $(\beta[a/x], \theta * \bar{\theta})$ is a pseudonatural transformation. Now define $\bar{\nu} : (\alpha[a/x], \eta * \bar{\eta}) \rightarrow (\beta[a/x], \theta * \bar{\theta})$ to be the modification given by $\bar{\nu}_K = 1_a$ and $\bar{\nu}_{K'} = \nu_{K'}$ for all $K' \neq K$. That $\bar{\nu}$ is a

modification follows immediately from the fact that ν is. By the inductive hypothesis we get that $(\beta[a/x], \theta * \bar{\theta}) \in \mathcal{M}(\Gamma, x: K \mid \phi)$. By taking $(a, \bar{\theta})$ to be the required data in the θ -coherent β -range of x it follows that $(\beta, \theta) \in \mathcal{M}(\Gamma \mid \exists x: K.\phi)$. The case of universal quantification follows exactly analogously. \square

Remark 4.12. Lemma 4.11 licenses us to consider only equivalence classes of pseudo-natural transformations up to modification as the extensions of our formulas. Indeed, for the purposes of this paper we are free to do so. (But note that taking such equivalence classes becomes more difficult in a type-theoretic metatheory.)

We conclude this section with a crucial example, that may allay a nagging suspicion: since all arrows in the interpretation of equality sorts are isomorphic to the identity, are we even able to assert the existence of non-trivial automorphisms? (This is a crucial design constraint for our system; for if we cannot even express that equality sorts can have non-reflexivity inhabitants, then nothing justifies the kind of groupoid interpretation we are here considering.)

Example 4.13. Consider the 1-signature \mathcal{L} with only one non-logical sort O of dimension 1. Define the (strict) \mathcal{L} -structure that takes O to $\mathbb{Z}/2$, with the latter regarded as a groupoid with a single object a and a single non-identity arrow $p: a \rightarrow a$. We want to show that

$$\mathbb{Z}/2 \models x: O \mid \exists q: x =_O^1 x. \forall \tau: r_O^1(q, x). \perp$$

To do so, it suffices to show that

$$\mathbb{Z}/2(\{x: O\} \mid \exists q: x =_O^1 x. \forall \tau: r_O^1(q, x). \perp)$$

is non-empty. Write α for the obvious pseudonatural transformation $\{x: O\} \Rightarrow \mathbb{Z}/2$ that sends x to a . We now have:

$$\begin{aligned} \mathbb{Z}/2(\{x: O\} \mid \exists q: x =_O^1 x. \exists \tau: r_O^1(q, x). \perp) &= \\ &= \{\alpha[\exists(\bar{q}, \bar{\eta}) \in \mathbb{Z}/2_\alpha[q] \text{ s.t. } a[\bar{q}/q] \in \mathbb{Z}/2(\{x, q\}: \forall \tau: r_O^1(q, x). \perp)]\} \\ &= \{\alpha[\exists(\bar{q}, \bar{\eta}) \forall(\bar{\tau}, \bar{\xi}) \in \mathbb{Z}/2_{\alpha[\bar{q}/q]}[\bar{\tau}] \text{ s.t. } \alpha[\bar{q}/q][\bar{\tau}/\tau] \in \emptyset]\} \end{aligned}$$

The last line above indicates that the required set is non-empty if there is a particular choice of \bar{q} and $\bar{\eta}$ that cannot be coherently extended to a pseudonatural transformation. So it suffices to show that there is indeed such a choice of \bar{q} and $\bar{\eta}$. To see this take

$$\bar{q} = \bar{\eta}_t = \bar{\eta}_s = p$$

where t and s denote the target and source morphisms in \mathcal{L} from $=_O^1$ to O . Similarly let ρ denote the unique morphism $r_O^1 \rightarrow =_O^1$. Now assume that there is an extension of this data into a pseudonatural transformation $(\alpha[\bar{q}/q][\bar{\tau}/\tau], \bar{\xi})$. The only possible choices for $\bar{\xi}_\rho$ are $(p, 1_a)$ and $(1_a, p)$. But since $t\rho = s\rho$ we must have $\bar{\xi}_{s\rho} = \bar{\xi}_{t\rho}$ which means that

$$\eta_t \circ t(\bar{\xi}_\rho) = \eta_s \circ s(\bar{\xi}_\rho)$$

So for any of the two choices of $\bar{\xi}_\rho$ we get $p \circ p = p$ which is a contradiction since p is its own inverse in $\mathbb{Z}/2$.

Remark 4.14. We see in the above example that falsity in our system is best understood as incoherence. In particular, the existence of non-identity (auto)morphisms in Example 4.13 is parsed as the non-existence of a coherent extension of a certain choice of data. So although every morphism will be isomorphic to the identity, not all of them will be coherently so. Our semantics is thus best understood as a semantics of *coherence*, not of *truth*. The two notions coincide for 0-logic but come apart for n -logic with $n \geq 1$. Satisfaction for n -logic is not to be understood merely as the existence of certain data, but as the *coherent* existence of certain data. This basic intuition is essential for the proof of completeness in the sequel.

5. PROOF SYSTEM FOR n -LOGIC

We now describe a proof system \mathcal{D} for n -logic. Since we want to work with possibly empty sorts as denotations (i.e. we do not want the inhabitation of sorts to be valid in our system) we have to “stratify” our formulas with respect to given contexts. The most convenient way of achieving this proof-theoretically is via a sequent calculus. On the other hand, as we shall see, our version of the J -rule forces us into at least the coherent fragment of first-order logic (i.e. formulas built out of the connectives \exists, \vee, \wedge). Since conjunction will therefore always be available to us, this justifies our decision to restrict ourselves to “singular” sequents.

Let \mathcal{L} be an n -signature. Then \mathcal{D} consists of the standard rules (given in the Appendix) of (intuitionistic or classical, coherent or full) first-order logic to which we add three rules. The first rule corresponds to the introduction rule for identity types in MLTT:

$$\frac{}{\Gamma, x: K \mid \theta \vdash \exists p: x =_K^1 x. \exists \tau: r_K^1(p, x). \top} \quad (\text{Eq-intro})$$

The second rule is the crucial J -rule, which corresponds to the elimination rule for identity types in MLTT:

$$\frac{\Gamma, x: K \mid \exists q_1: x =_K^1 x. r_K^1(q_1) \wedge \theta[x/y, q_1/p] \vdash \exists q_2: x =_K^1 x. r_K^1(q_2) \wedge \phi[x/y, q_2/p]}{\Gamma, x: K, y: K, p: x =_K^1 y \mid \theta \vdash \phi} \quad (\text{J})$$

Remark 5.1. This is where the decision to include reflexivity sorts of dimension -2 pays off. It allows us to have a uniform presentation of the rules for equality sorts without splitting them into “propositional” and “non-propositional”, which would force us into two separate sets of rules (one for “top-level” equalities and one for the rest). What we can do instead is prove that “top-level” equalities behave like propositions, as indeed we shall do in Proposition 5.3 below.

Finally, we add the following rule expressing the fact that “reflexivity is unique up to equality one level up”:

$$\frac{}{\Gamma, x: K, p: x =_K^1 x, q: x =_K^1 x \mid r_K^1(p) \wedge r_K^1(q) \vdash \exists p =_K^2 q} \quad (\text{R})$$

Remark 5.2. In our presentation above we have assumed that every sequent is well-formed and that the rules are only instantiated for appropriate sorts (e.g. (Eq-intro) is instantiated only for K with $d(K) < -1$). Furthermore, we have suppressed all variable dependencies that are not directly relevant to the rule in question and have also assumed that sorts K as they appear could themselves denote equality sorts (in which case we understand $=_K^1$ as an abbreviation for $=_{K'}^{i+1}$ where K is $=_{K'}^i$). Finally, the notation $\phi[x/y, q/p]$ in (J) denotes the formula $\delta(\phi)$ obtained by substitution along the contraction morphism described in Example 1.10.

If we so choose we can certainly add the law of the excluded middle (LEM) as an axiom. When we do so we will denote the corresponding proof system by \mathcal{D}^{cl} . When what we say applies to both \mathcal{D} and \mathcal{D}^{cl} we will use the notation $\mathcal{D}^{(\text{cl})}$.

We note that the (J) rule above corresponds to what in MLTT would be called *strong* Id-elimination since θ behaves like a contextual parameter that may itself depend on the variables $\{x, y, p\}$. In the presence of Π -types, strong Id-elimination is equivalent to the usual form (without a contextual parameter). Similarly, in the presence of universal quantification (J) is equivalent to the more recognizable rule

$$\frac{\Gamma, x: K \mid \top \vdash \exists q: x =_K^1 x. \exists \tau: r_K^1(q, x). \phi[x/y, q/p]}{\Gamma, x: K, y: K, p: x =_K^1 y \mid \top \vdash \phi} \quad (\text{J}')$$

The (J)-rule can be used to prove that “top-level” equality behaves exactly like a proposition. This fact means that the “top-level” equality sort of any other sort $K \in \mathcal{L}$ (of non-maximal level) will behave like a proposition, the “second-from-top” equality sort will behave like a set etc. Our proof system thus captures the crucial element that our (homotopy) semantics demands: that sorts of dimension m behave like m -types. The following proposition registers this fact.

Proposition 5.3 (“Top-level equality is propositional”). *For any $K \in \text{ob}\mathcal{L}$ with dimension $d(K) = 0$ and any \mathcal{L} -formula ϕ in context $\Gamma, x: K, y: K$ the following rule (called “Lawvere equality” in [Jac99]) is derivable:*

$$\frac{\Gamma, x: K \mid \theta \vdash \phi[x/y]}{\Gamma, x: K, y: K \mid \theta, \exists p: x =_K^1 y. \top \vdash \phi} \quad (\text{L-eq})$$

Proof. Since K is assumed to be of dimension 0, we have $d(=_K^1) = -1$ and therefore for any formula ϕ the substitution $\phi[x/y]$ is well-defined. We then have:

$$\begin{array}{c} \frac{\Gamma, x: K \mid \theta \vdash \phi[x/y]}{\Gamma, x: K \mid \theta \vdash \phi[x/y] \wedge \exists q: x =_K^1 x. r_K^1(q)} \quad (\text{Eq-intro}) \\ \frac{\Gamma, x: K \mid \theta \vdash \phi[x/y] \wedge \exists q: x =_K^1 x. r_K^1(q)}{\Gamma, x: K \mid \theta \vdash \exists q: x =_K^1 x. (\phi[x/y] \wedge r_K^1(q))} \quad (\text{Frob}) \\ \frac{\Gamma, x: K, y: K, p: x =_K^1 y \mid \theta \vdash \phi}{\Gamma, x: K, y: K \mid \theta, \exists p: x =_K^1 y. \top \vdash \phi} \quad (\text{J}) \\ \frac{\Gamma, x: K, y: K \mid \theta, \exists p: x =_K^1 y. \top \vdash \phi}{\Gamma, x: K, y: K \mid \theta, \exists p: x =_K^1 y. \top \vdash \phi} \quad (\exists\text{-intro}) \end{array}$$

In applying (\exists -intro) and (Frob) we have assumed that p does not appear in ϕ , as indeed we can do without loss of generality. \square

One gets transport for free in HoTT (cf. [Uni13], p.91) as a consequence of the J -rule for identity types of the ambient type theory. By a similar argument, we also get the relevant version of transport in our setting.

Proposition 5.4 (“Transport”). *For any \mathcal{L} -formula ϕ in context $\{\Gamma, x: K\}$ the sequent*

$$\Gamma, x: K, y: K, p: x =_K^1 y \mid \phi \vdash \phi[y/x]$$

is derivable.

Proof. By (iden) we know that the sequent

$$\Gamma, x: K \mid \phi[x/y] \vdash \phi[y/x][x/y]$$

is derivable (from no premises) since $\phi[x/y] \equiv \phi[y/x][x/y]$. But then we can just repeat the (first three steps of) the derivation in the proof of Proposition 5.3 with $\theta \equiv \phi[x/y]$ and $\phi \equiv \phi[y/x]$ to get the desired result. \square

We may now define the notion of n -theory in the usual manner: an \mathcal{L} - n -theory is given by (the $\mathcal{D}^{(\text{cl})}$ -closure of) a set of \mathcal{L} - n -sentences (its axioms). As usual we will write $\mathcal{M} \models \mathbb{T}$ for an \mathcal{L} - n -structure \mathcal{M} that satisfies all the axioms of \mathbb{T} . Following [Uni13], we write LEM for the following type in UF:

$$\prod_{A: \mathbf{Prop}_u} A + \neg A$$

For any \mathcal{L} - n -theory \mathbb{T} we write $\mathbb{T} \vdash \phi$ (resp. $\mathbb{T} \vdash_{\text{cl}} \phi$) to denote that ϕ is derivable in \mathcal{D} (resp. $\mathcal{D}^{(\text{cl})}$) from \mathbb{T} . We write $\mathbb{T} \models \phi$ (resp. $\mathbb{T} \models_{\text{cl}} \phi$) to denote that ϕ is true in all models of \mathbb{T} in HoTT (resp. HoTT+LEM) and similarly for the set-theoretic semantics. At this point we have obtained all the components traditionally required of a logic: a syntax, a semantics and a proof system. We may thus begin to investigate the relationship between these components, as indeed we do in the next section.

Remark 5.5. One might wonder whether the class of (homotopy) models we are considering for n -logic is too wide. In particular, since we appear not to be making any use of the full structure of identity types one might wonder whether our n -logic can take semantics where *every* type is interpreted as a set (i.e. a 0-type) much as in Makkai’s original formulation of the semantics of FOLDS. This is not the case. n -logic does in fact have the expressive power to force a theory to have only models whose ground sorts are n -types. As the simplest possible illustration, take the case where $n = 1$ and \mathcal{L} is the same signature as in Example 4.13. Consider the \mathcal{L} -1-theory \mathbb{T}_O consisting of the single axiom

$$\phi \equiv \forall x, y: O. \exists p, q: x =_O^1 y. \neg(p =_O^2 q)$$

with the obvious abbreviations. Every model of \mathbb{T}_O where O is interpreted as an h -set (resp. discrete groupoid) falsifies ϕ . But \mathbb{T}_O is satisfiable: simply take an \mathcal{L} -1-structure where O is interpreted as a (proper) 1-type (resp. groupoid). Assuming the soundness results of Section 6 this means that ϕ cannot be disproved by \mathcal{D} even though it is not satisfied in any set-model of \mathbb{T}_O . As such, set models are not sufficient to describe provability for 1-logic. Exactly analogously, we can see that $(n - 1)$ -type models are not sufficient to describe provability for n -logic.

Remark 5.6. Even if n -logic is expressive enough to go beyond set-theoretic semantics, one may still suspect that $\mathcal{D}^{(cl)}$ is much too impoverished to capture enough of the complexity of identity types in MLTT to prove a completeness theorem. Recall however, that the syntax of n -logic does not include any closed terms and the actual equality sorts have already been introduced through the process of globular completion. Therefore, the rules for identity types whose analogues do not appear in $\mathcal{D}^{(cl)}$ (i.e. **Id-intro** and **Id-comp**) would not sensibly translate to anything that we may wish to do in \mathcal{D} . (**Id-intro** has already been applied in the formation of formulas and **Id-comp** applies to closed terms which we do not have). This should give some initial plausibility that the (homotopy) semantics we have outlined are complete with respect to \mathcal{D} since there is nothing else in MLTT that one can do with h -props that cannot be done with the (suitable translates of the) rules of \mathcal{D} .

6. SOUNDNESS

We will now prove soundness for the rules of \mathcal{D} with respect to both our semantics. We begin with the homotopy semantics of Section 3.

Theorem 6.1 (Soundness for homotopy semantics). *If $\mathbb{T} \vdash_{(cl)} \phi$ then $\mathbb{T} \models_{(cl)} \phi$.*

Proof. All the traditional rules of \mathcal{D} have direct analogues in MLTT and so the proof proceeds without difficulties by induction on the complexity of derivations. This means that we take each particular rule in the deductive system and show that there is a derivation from the translation of the top line to the translation of the bottom line. There is one minor subtlety. Axioms in \mathcal{D} are stated by starting from an empty line and then producing a formula. For example, in \mathcal{D} we have the following axiom

$$\frac{}{\Gamma \mid \phi \vdash \phi} \text{ iden}$$

There is a certain amount of information suppressed in stating such an axiom, namely that the given sequent is well-formed. In translating an axiom like (iden) we will therefore translate the “empty” set of formulas above as the judgement in HoTT stating that the types involved in the translation of the formula below are well-formed. (This is essentially the price we pay when we interpret a proof-irrelevant system into a proof-relevant one.) So in translating (iden) we need to show that there is a HoTT-derivation from the judgment

$$\Delta^{\mathcal{M}}, \Gamma^{\mathcal{M}} \vdash \phi^{\mathcal{M}} : \mathcal{U}$$

to the judgment

$$\Delta^{\mathcal{M}} \vdash s : \prod_{\vec{x} : \Gamma^{\mathcal{M}}} \phi^{\mathcal{M}} \rightarrow \phi^{\mathcal{M}}$$

for some s and for any \mathcal{M} . In a somewhat abbreviated form this goes as follows

$$\frac{\frac{\frac{\Delta^{\mathcal{M}}, \Gamma^{\mathcal{M}} \vdash \phi^{\mathcal{M}} : \mathcal{U}}{\Delta^{\mathcal{M}}, \Gamma^{\mathcal{M}} \vdash \phi^{\mathcal{M}} \rightarrow \phi^{\mathcal{M}} : \mathcal{U}} \quad \text{II-form, wkg}}{\Delta^{\mathcal{M}} \vdash \prod_{\vec{x} : \Gamma^{\mathcal{M}}} \phi^{\mathcal{M}} \rightarrow \phi^{\mathcal{M}}} \quad \text{II-form}} \quad \text{II-intro}$$

where the term $\lambda \vec{y}. \vec{y}$ has been produced by applying II-intro to

$$\Delta^{\mathcal{M}}, \Gamma^{\mathcal{M}}, y : \phi^{\mathcal{M}} \vdash y : \phi^{\mathcal{M}}$$

We do the same for other axioms. Since the languages we are considering are purely relational and there are no closed terms, the substitution rule follows immediately without any complications. As for the cut rule, given terms

$$\Delta^{\mathcal{M}} \vdash \eta : \prod_{\vec{x} : \Gamma^{\mathcal{M}}} \phi^{\mathcal{M}} \rightarrow \psi^{\mathcal{M}}$$

and

$$\Delta^{\mathcal{M}} \vdash \xi : \prod_{\vec{x} : \Gamma^{\mathcal{M}}} \psi^{\mathcal{M}} \rightarrow \chi^{\mathcal{M}}$$

we can define a term

$$\Delta^{\mathcal{M}} \vdash \lambda \vec{x}. (\lambda y : \phi^{\mathcal{M}}. \xi(\vec{x})(\eta(\vec{x})(y))) : \prod_{\vec{x} : \Gamma^{\mathcal{M}}} \phi^{\mathcal{M}} \rightarrow \chi^{\mathcal{M}}$$

The rest of the structural rules follow just as easily.

The logical rules follow by interpreting the connectives in the manner of Section 3. We will do the case of existential quantification as an illustration. The relevant rule is the following

$$(\exists) \frac{\Gamma, x : K \mid \phi \vdash \psi}{\Gamma \mid \exists x : K. \phi \vdash \psi} \quad x \notin |\Gamma|$$

So suppose we have derived the judgement

$$\Delta^{\mathcal{M}} \vdash \eta : \prod_{\substack{\vec{x} : \Gamma^{\mathcal{M}} \\ y : K^{\mathcal{M}}}} \phi^{\mathcal{M}} \rightarrow \psi^{\mathcal{M}}$$

Then we can define the following term

$$\Delta^{\mathcal{M}} \vdash \xi \equiv \lambda \vec{x}. (\lambda \langle y, p \rangle. \eta(\vec{x}, y)(p)) : \prod_{\vec{x} : \Gamma^{\mathcal{M}}} \sum_{y : K^{\mathcal{M}}} \phi^{\mathcal{M}} \rightarrow \psi^{\mathcal{M}}$$

But since $\psi^{\mathcal{M}}$ will be an h -prop for any substitution instance of its free variables, by the universal property of the propositional truncation we get a term

$$\Delta^{\mathcal{M}} \vdash \xi \equiv \lambda \vec{x}. ||(\lambda \langle y, p \rangle. \eta(\vec{x}, y)(p))|| : \prod_{\vec{x} : \Gamma^{\mathcal{M}}} || \sum_{y : K^{\mathcal{M}}} \phi^{\mathcal{M}} || \rightarrow \psi^{\mathcal{M}}$$

This is exactly the translation of the (satisfaction of the) bottom sequent in (\exists) . The rest of the logical rules follow similarly, employing the universal property of the propositional truncation when needed.

So it remains to prove soundness for (Eq-intro), (J) and (R). For (Eq-intro) we have that

$$\Delta^{\mathcal{M}} \vdash \langle \mathbf{refl}_x, \mathbf{refl}_{\mathbf{refl}_x} \rangle : \parallel \sum_{p: \text{Id}_{K^{\mathcal{M}}}(x, x)} \text{Id}_{\text{Id}_{K^{\mathcal{M}}}(x, x)}(p, \mathbf{refl}_x) \parallel$$

is derivable which gives us the required result. For (J), given the availability of Π -types in our metatheory, we will prove (J') for simplicity since the main idea of the argument is exactly the same. So suppose we are given:

$$\mathcal{M} \models \Gamma, x: K \mid \top \vdash \exists q: x =_K x. \exists \tau: r_K(q). \phi[x/y, q/p]$$

This means that the following judgement is derivable in HoTT:

$$\Delta^{\mathcal{M}} \vdash \eta: \prod_{\substack{\vec{z}: \Gamma^{\mathcal{M}} \\ x: K^{\mathcal{M}}}} \parallel \sum_{q: \text{Id}_{K^{\mathcal{M}}}(x, x)} \text{Id}_{\text{Id}_{K^{\mathcal{M}}}(x, x)}(q, \mathbf{refl}_x) \times \phi^{\mathcal{M}}[x/y, q/p] \parallel$$

Since $\phi^{\mathcal{M}}$ is an h -prop we can remove it from the propositional truncation to obtain that the following judgment is derivable (where once again denote the term by η):

$$\Delta^{\mathcal{M}} \vdash \eta: \prod_{\substack{\vec{z}: \Gamma^{\mathcal{M}} \\ x: K^{\mathcal{M}}}} \parallel \sum_{q: \text{Id}_{K^{\mathcal{M}}}(x, x)} \text{Id}_{\text{Id}_{K^{\mathcal{M}}}(x, x)}(q, \mathbf{refl}_x) \parallel \times \phi^{\mathcal{M}}[x/y, q/p]$$

Now we can define a term

$$\xi: \prod_{\substack{\vec{z}: \Gamma^{\mathcal{M}} \\ x: K^{\mathcal{M}}}} \sum_{q: \text{Id}_{K^{\mathcal{M}}}(x, x)} \text{Id}_{\text{Id}_{K^{\mathcal{M}}}(x, x)}(q, \mathbf{refl}_x) \times \phi^{\mathcal{M}}[x/y, q/p] \rightarrow \phi^{\mathcal{M}}[x/y, q/\mathbf{refl}_x]$$

by

$$\xi \equiv \lambda \vec{z}, x. \lambda w. \pi_1(\mathbf{pr}_2(w))^*(\pi_2(\mathbf{pr}_2(w)))$$

and since $\phi^{\mathcal{M}}$ will always be an h -prop once again by the universal property of the propositional truncation this gives us a term

$$\bar{\xi}: \prod_{\substack{\vec{z}: \Gamma^{\mathcal{M}} \\ x: K^{\mathcal{M}}}} \parallel \sum_{q: \text{Id}_{K^{\mathcal{M}}}(x, x)} \text{Id}_{\text{Id}_{K^{\mathcal{M}}}(x, x)}(q, \mathbf{refl}_x) \parallel \times \phi^{\mathcal{M}}[x/y, q/p] \rightarrow \phi^{\mathcal{M}}[x/y, q/\mathbf{refl}_x]$$

By using η we thus get a term

$$\Delta^{\mathcal{M}} \vdash \bar{\eta}: \prod_{\substack{\vec{z}: \Gamma^{\mathcal{M}} \\ x: K^{\mathcal{M}}}} \phi^{\mathcal{M}}[x/y, \mathbf{refl}_x]$$

But then by the induction rule for identity types (i.e. Id-elim internalized) we get

$$\Delta^{\mathcal{M}} \vdash \mathbf{ind}_{=}(\bar{\eta}): \prod_{\substack{\vec{w}: (\Gamma \setminus \{y, p\})^{\mathcal{M}} \\ x, y: K^{\mathcal{M}} \\ p: \text{Id}_{K^{\mathcal{M}}}(x, y)}} \phi^{\mathcal{M}}$$

which gives us the desired result. For (R) we may simply apply the well-known transitivity of identity types, which immediately gives us the desired result. Finally, the soundness of LEM in HoTT+LEM is evident. \square

We now also prove soundness for the set-theoretic semantics outlined in Section 4. The main difficulty is once again in showing the soundness of the (J)-rule. In the proof below this is achieved by arguing that the extension of any formula with a path in its free variables is fully (and coherently) determined by individual objects in the groupoid in which that path is interpreted.

Theorem 6.2 (Soundness for set-theoretic semantics of 1-logic). *If $\mathbb{T} \vdash_{(cl)} \phi$ then $\mathbb{T} \models_{(cl)} \phi$.*

Proof. The structural rules and the rules for the connectives follow immediately just as in normal first-order logic. (R) and (Eq-intro) also follow immediately since any sort of dimension 1 is interpreted as a groupoid and will by definition contain unique (up to equality) identity arrows. As before, the only non-trivial case is (J). We will consider the case of (J') since the core of the argument is exactly the same and keeping track of parameters unnecessarily obfuscates it. Furthermore, since the formulas to which (J') meaningfully applies will depend on sorts of dimension 1 and hence will not depend on other sorts below them, it suffices to consider the case where there is no ambient context Γ . Therefore, it suffices to prove the soundness of the following simplified rule

$$\frac{x: K \mid \top \vdash \exists q: x =_K^1 x. \exists \tau: r_K^1(q, x). \phi[x/y, q/p]}{x: K, y: K, p: x =_K^1 y \mid \top \vdash \phi} \quad (\text{J}'')$$

where $\phi[x/y, q/p]$ denotes the formula $\delta(\phi)$ obtained from the change-of-variables

$\delta: \{x, y, p\} \Rightarrow \{x, q\}$ described in Example 1.10.

So suppose we are given an \mathcal{L} -structure \mathcal{M} such that

$$(*) \quad \mathcal{M}(\{x: K\} \mid \top) \subseteq \mathcal{M}(\{x: K\} \mid \exists q: x =_K^1 x. \exists \tau: r_K^1(q, x). \phi[x/y, q/p])$$

We need to show that $(*)$ implies

$$(**) \quad \mathcal{M}(\{x, y: K, p: x =_K^1 y\} \mid \top) \subseteq \mathcal{M}(\{\Gamma, x, y: K, p: x =_K^1 y\} \mid \phi)$$

Assume for simplicity that \mathcal{L} is strict. Take an arbitrary pseudonatural transformation $(\alpha, \eta): \{x, y, p\} \Rightarrow \mathcal{M}$ which consists of the following data

$$a = \alpha_K(x) \quad b = \alpha_K(y)$$

$$\alpha_{=_K^1}(p) = a' \xrightarrow{\bar{p}} b'$$

$$\eta_s: a' \rightarrow a \quad \eta_t: b' \rightarrow b$$

where as usual s and t denote the source and target maps of the equality sort $=_K^1$. Consider now the pseudonatural transformation $[a/x]: \{x\} \Rightarrow \mathcal{M}$ that takes x to a (and contains trivial coherence data). By $(*)$ we know that

$$(1) \quad [a/x] \in \mathcal{M}(\{x: K\} \mid \exists q: x =_K^1 x. \exists \tau: r_K^1(q, x). \phi[x/y, q/p])$$

In particular this means that there exists $\bar{q}: a'' \rightarrow b''$, $\theta_s: a'' \rightarrow a$, $\theta_t: b'' \rightarrow a$ such that, by Lemma 4.10, we get

$$(\beta, \theta) \equiv ([a/x][a/y][\bar{q}/p], \{\theta_s, \theta_t\}) \in \mathcal{M}(\{x, y, p\} \mid \phi)$$

We now claim that $(\alpha, \eta) \cong (\beta, \theta)$. To define a modification $\nu: (\alpha, \eta) \rightarrow (\beta, \theta)$ we need arrows

$$\begin{aligned} (\nu_K)_x &: \alpha_K(x) = a \rightarrow a = \beta_K(x) \\ (\nu_K)_y &: \alpha_K(y) = b \rightarrow b = \beta_K(x) \\ (\nu_{=1_K})_p &: \alpha_{=1_K}(p) = \bar{p} \rightarrow \bar{q} = \beta_{=1_K}(p) \end{aligned}$$

satisfying the required coherence condition

$$\theta_f \circ \mathcal{M}(f)((\nu_{=1_K})_p) = (\nu_K)_{\Gamma(f)(p)} \circ \eta_f$$

where we have used Γ as notation for the associated functor of the given context $\{x, y, p\}$ and where $f \in \{s_K^1, t_K^1\}$. We fill this data as follows:

$$\begin{aligned} (\nu_K)_x &\equiv \theta_{s\rho} \\ (\nu_K)_y &\equiv \theta_{t\rho} \circ \eta_s \circ (\eta_t \circ \bar{p})^{-1} \\ (\nu_{=1_K})_p &\equiv ((\theta_\rho)_1 \circ \eta_s, (\theta_\rho)_2 \circ \eta_s \circ \bar{p}^{-1}) \end{aligned}$$

where θ_ρ , $\theta_{t\rho}$ and $\theta_{s\rho}$ are given by the fact that we know by (1) that there exists a coherent isomorphism from 1_a to \bar{q} . It remains to check that ν satisfies the required coherence conditions. Firstly, for s , we have

$$\begin{aligned} \theta_s \circ \mathcal{M}(s)((\nu_{=1_K})_p) &= \theta_s \circ (\theta_\rho)_1 \circ \eta_s \\ &= \theta_s \circ \mathcal{M}(s)(\theta_\rho) \circ \eta_s \\ &= \theta_{s\rho} \circ \eta_s \\ &= (\nu_K)_x \circ \eta_s \end{aligned}$$

Finally, for t , we have

$$\begin{aligned} \theta_t \circ \mathcal{M}(t)((\nu_{=1_K})_p) &= \theta_t \circ (\theta_\rho)_2 \circ \eta_s \circ \bar{p}^{-1} \\ &= \theta_t \circ \mathcal{M}(t)(\theta_\rho) \circ \eta_s \circ \bar{p}^{-1} \\ &= \theta_{t\rho} \circ \eta_s \circ \bar{p}^{-1} \\ &= \theta_{t\rho} \circ \eta_s \circ (\eta_t \circ \bar{p})^{-1} \circ \eta_t \\ &= (\nu_K)_y \circ \eta_t \end{aligned}$$

This establishes that ν is a modification and therefore that $(\alpha, \eta) \cong (\beta, \theta)$. By Lemma 4.11 this means that $(\alpha, \eta) \in \mathcal{M}(\{x, y, p\} \mid \phi)$ and therefore that

$$\mathcal{M}(\{x, y, p\} \mid \phi) = \mathbf{pNat}(\{x, y, p\}, \mathcal{M})$$

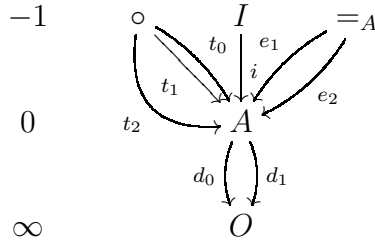
which is exactly (**). Finally, as noted in Remark 4.8, LEM is clearly sound since we are working in a classical set theory. \square

7. EXAMPLES AND APPLICATIONS

FOLDS was invented as a systematic way of avoiding the use of equalities that are irrelevant for the structures of interest, e.g. equality between objects when we care about categories only up to equivalence. On the other hand, n -logic represents

a reversal of this idea, since we are re-introducing equalities as logical sorts with a fixed interpretation. But we don't have to do this in order to get some traction out of FOLDS as a proof-irrelevant syntax with a natural semantics in HoTT. Indeed, normal FOLDS can be employed to axiomatize structures in UF as long as we allow ourselves to fix the interpretation of certain sorts as identity types in HoTT. Another (equivalent) way to achieve this is simply to extend the dimension functions d to $\mathbb{Z}_{\geq -2} \cup \{\infty\}$, i.e. to allow for certain sorts to be types of arbitrarily large dimension. More precisely, we will call (\mathcal{L}, d) an ∞ -signature if it is a FOLDS-signature where d is allowed to take values in $\mathbb{Z}_{\geq -2} \cup \{\infty\}$ and where the interpretation of a sort K with $d(K) = \infty$ is an arbitrary type $K^{\mathcal{M}}: \mathcal{U}$.

Now, let \mathcal{L}_{cat} denote the following ∞ -signature



subject to the relations

$$d_0 t_0 = d_0 t_2, d_1 t_1 = d_1 t_2, d_0 t_1 = d_1 t_0$$

$$d_0 i = d_1 i$$

$$d_0 e_1 = d_0 e_2, d_1 e_1 = d_1 e_2$$

The \mathcal{L}_{cat} -theory of categories \mathbb{T}_{cat} consists of the following axioms:

(1) (Existence of identities)

$$\forall x: O. \exists i: A(x, x). \exists \sigma: I(i, x, x). \top$$

(2) (Functionality of composition-1)

$$\forall x, y, z: O. \forall f: A(x, y). \forall g: A(y, z). \exists h: A(x, z). \exists \tau: \circ(f, g, h, x, y, z). \top$$

(3) (Functionality of Composition-2)

$$\forall x, y, z: O. \forall f: A(x, y). \forall g: A(y, z). \forall h, h': A(x, z). \forall \tau_1: \circ(f, g, h). \forall \tau_2: \circ(f, g, h').$$

$$\exists \epsilon: =_A(h, h', x, z)$$

(4) (Associativity)

$$\forall x, y, z, w: O. \forall f: A(x, y). \forall g: A(y, z). \forall h: A(z, w). \forall i: A(x, z). \forall j: A(x, w).$$

$$\forall k: A(y, w). \forall \tau_1: \circ(f, g, i, x, y, z). \forall \tau_2: \circ(i, h, j, x, z, w). \forall \tau_3: \circ(g, h, k, y, z, w).$$

$$\exists \tau_4: \circ(f, k, j, x, y, w). \top$$

(5) (Uniqueness of identity)

$$\forall x: O. \forall i, j: A(x, x). \forall \sigma_1: I(i, x, x). \forall \sigma_2: I(j, x, x). \exists \epsilon: =_A(i, j, x, x). \top$$

(6) (Right unit)

$$\forall x, y: O. \forall i: A(x, x). \forall g: A(x, y). \forall \sigma: I(i, x, x). \exists \tau: \circ(i, g, g, x, x, y). \top$$

(7) (Left unit)

$$\forall x, y: O. \forall i: A(y, y). \forall f: A(x, y). \forall \phi: I(i, y, y). \exists \tau: \circ(f, i, f, x, y, y). \top$$

Using our homotopy semantics we get that an \mathcal{L}_{cat} -structure consists of the following data:

- A type $O: \mathcal{U}$
- A term $A: O \rightarrow O \rightarrow \mathbf{Set}_{\mathcal{U}}$
- A term $I: \prod_{a: O} A(a, a) \rightarrow \mathbf{Prop}_{\mathcal{U}}$
- A term $\circ: \prod_{x, y, z: C} A(x, y) \rightarrow A(y, z) \rightarrow A(x, z) \rightarrow \mathbf{Prop}_{\mathcal{U}}$

We can now translate the axioms of \mathbb{T}_{cat} into types in HoTT for an arbitrary model \mathcal{M} of \mathbb{T}_{cat} . We will list them in order, writing $=$ for the identity on $A^{\mathcal{M}}$ and omitting \mathcal{M} from superscripts:

$$\begin{array}{ll}
(T_1) \prod_{x: O} \prod_{i: A(x, x)} \sum_{i: A(x, x)} I(i, x, x) & (T_5) \prod_{x: O} i = j \\
(T_2) \prod_{x, y, z: O} \prod_{f: A(x, y)} \prod_{g: A(y, z)} \sum_{h: A(x, z)} \circ(f, g, h, x, y, z) & (T_6) \prod_{x, y: O} \prod_{i: A(x, x)} \prod_{g: A(x, y)} \prod_{\phi: I(i, x, x)} \prod_{\psi: I(j, x, x)} \circ(i, g, g, x, x, y) \\
(T_3) \prod_{x, y, z: O} \prod_{f: A(x, y)} \prod_{g: A(y, z)} \prod_{h, h': A(x, z)} h = h' & (T_7) \prod_{x, y: O} \prod_{i: A(y, y)} \prod_{f: A(x, y)} \prod_{\phi: I(i, y, y)} \circ(f, i, f, x, y, y) \\
(T_4) \prod_{x, y, z, w: O} \prod_{\tau_1: \circ(f, g, h, x, y, z)} \prod_{\tau_2: \circ(f, g, h', x, y, z)} \prod_{\tau_3: \circ(g, h, k, y, z, w)} \circ(f, k, j, x, y, w) &
\end{array}$$

We thus obtain:

$$\begin{aligned}
\mathbf{Mod}(\mathbb{T}_{\text{cat}}) \equiv & \sum_{\substack{O: \mathcal{U} \\ A: O \rightarrow O \rightarrow \mathbf{Set}_{\mathcal{U}} \\ I: \prod_{a: O} A(a, a) \rightarrow \mathbf{Prop}_{\mathcal{U}} \\ \circ: \prod_{a, b, c: C} A(a, b) \rightarrow A(b, c) \rightarrow A(a, c) \rightarrow \mathbf{Prop}_{\mathcal{U}}}} T_1 \times T_2 \times T_3 \times T_4 \times T_5 \times T_6 \times T_7
\end{aligned}$$

Following [Uni13], a *precategory* is defined by the following data:

- (1) A type $C: \mathcal{U}$ (“objects”)
- (2) A dependent type $\text{Hom}_C: C \rightarrow C \rightarrow \mathbf{Set}_{\mathcal{U}}$ (“Hom-sets”)
- (3) $1: \prod_{a: C} \text{Hom}_C(a, a)$
- (4) $\circ: \prod_{a, b, c: C} \text{Hom}_C(a, b) \rightarrow \text{Hom}_C(b, c) \rightarrow \text{Hom}_C(a, c)$
- (5) **assoc**: $\prod_{a, b, c, d: C} \prod_{f: \text{Hom}_C(a, b)} \prod_{g: \text{Hom}_C(b, c)} \prod_{h: \text{Hom}_C(c, d)} h \circ (g \circ f) = (h \circ g) \circ f$
- (6) **ident**: $\prod_{a, b: C} \prod_{f: \text{Hom}_C(a, b)} (f \circ 1_a = f) \times (1_b \circ f = f)$

We can now show that precategories are “ ∞ -elementary” in the sense that they are axiomatizable, up to equivalence, by \mathbb{T}_{cat} over the ∞ -signature \mathcal{L}_{cat} . In what follows

below we will be making free use of the HoTT version of the Axiom of Unique Choice (AUC) ([Uni13], Corollary 3.9.2).

Proposition 7.1. $\mathbf{PreCat} \simeq \mathbf{Mod}(\mathbb{T}_{cat})$

Proof. The proof boils down to proving, using AUC, that an axiomatization of a category in terms of a relation of composition is equivalent to the usual axiomatization in terms of an operation of composition. First we define a function

$$p: \mathbf{Mod}(\mathbb{T}_{cat}) \rightarrow \mathbf{PreCat}$$

So let $C: \mathbf{Mod}(\mathbb{T}_{cat})$ and write t_i for the inhabitants of each axiom T_i that is part of the data of C . We need to provide the data for conditions (1)-(6) in the definition of precategories. We are given O and A and those immediately take care of conditions (1) and (2). For condition (3) we first observe that $\sum_{i: A(x,x)} I(i, x, x)$ is a mere proposition for any $x: O$. For suppose that (i, ϕ) and (j, ψ) are two terms of type $\sum_{i: A(x,x)} I(i, x, x)$. To show that $(i, \phi) = (j, \psi)$ it suffices to show that there is $p: i = j$ and that $p_*(\phi) = \psi$. By applying t_5 to the data $\langle x, i, j, \phi, \psi \rangle$ we get a proof that $i = j$, i.e. a term $p: i = j$. Clearly, since $I(i, x, x)$ and $I(j, x, x)$ are mere propositions, we also get that $\psi = p_*(\phi)$ and therefore we get that $(i, \phi) = (j, \psi)$ and therefore that the type $\sum_{i: A(x,x)} I(i, x, x)$ is a mere proposition. By AUC and t_1 we get a term

$$u: \prod_{x: O} \sum_{i: A(x,x)} I(i, x, x)$$

Thus we can define, for each $x: O$, the following term

$$1_x =_{\text{def}} \mathbf{pr}_1(u_x): A(x, x)$$

and thus we obtain a term

$$1^C =_{\text{def}} \lambda x. 1_x: \prod_{x: O} A(x, x)$$

as required by condition (3). Condition (4) follows similarly and we omit the details. For condition (5), let $x, y, z, w: O$ and $f: A(x, y), g: A(y, z)$ and $h: A(z, w)$. Now let

$$r =_{\text{def}} (t_4)_{x,y,z,w,f,g,h,g \circ^C f, h \circ^C (g \circ^C f), h \circ^C g}: \prod_{\substack{\tau_1: \circ(f, g, g \circ^C f, x, y, z) \\ \tau_2: \circ(g \circ^C f, h, h \circ^C (g \circ^C f), x, z, w) \\ \tau_3: \circ(g, h, h \circ^C g, y, z, w)}} \circ(f, h \circ^C g, h \circ^C (g \circ^C f), x, y, w)$$

We can then define

$$\begin{aligned} p_1 &=_{\text{def}} \mathbf{pr}_2(c_{x,y,z,f,g}) \\ p_2 &=_{\text{def}} \mathbf{pr}_2(c_{x,z,w,g \circ^C f,h}) \\ p_3 &=_{\text{def}} \mathbf{pr}_2(c_{y,z,w,g,h}) \end{aligned}$$

and thus $r_{p_1, p_2, p_3}: \circ(f, h \circ^C g, h \circ^C (g \circ^C f), x, y, w)$. But by definition we have a term $\pi: \circ(f, h \circ^C g, (h \circ^C g) \circ^C f, x, y, w)$ and therefore we get a term

$$(t_{11})_{x,y,w,f,f,h \circ^C g, h \circ^C g, \mathbf{refl}_f, \mathbf{refl}_{h \circ^C g}, h \circ^C (g \circ^C f), (h \circ^C g) \circ^C f, r_{p_1, p_2, p_3}, \pi}: h \circ^C (g \circ^C f) = (h \circ^C g) \circ^C f$$

Thus we can define

$$\text{assoc}_{x,y,z,w,f,g,h}^C =_{\text{def}} (t_4)_{x,y,w,f,f,h \circ^C g, h \circ^C g, \text{refl}_f, \text{refl}_{h \circ^C g}, h \circ^C (g \circ^C f), (h \circ^C g) \circ^C f, r_{p_1, p_2, p_3}, \pi}$$

and this gives us the section

$$\text{assoc}^C: \prod_{a,b,c,d: C} \prod_{\substack{f: \text{Hom}_C(a,b) \\ g: \text{Hom}_C(b,c) \\ h: \text{Hom}_C(c,d)}} h \circ (g \circ f) = (h \circ g) \circ f$$

as required by condition (5). Condition (6) follows similarly. So we can now write $p(C)$ for the precategory given by the data

$$(O, A, 1^C, \circ^C, \text{assoc}^C, \text{idl}^C, \text{idr}^C)$$

with the notation as in the proof of Proposition 7.1.

Conversely, we need to define a function

$$q: \mathbf{PreCat} \rightarrow \mathbf{Mod}(\mathbb{T}_{\text{cat}})$$

So let C be a precategory given by the data

$$(C, \text{Hom}, 1, \circ, \text{assoc}, \text{idl}, \text{idr})$$

Given 1 we know that for each $x: C$, $\text{Hom}(x, x)$ is inhabited since $1_x: \text{Hom}(x, x)$. Thus we can define

$$I_C =_{\text{def}} \lambda x. (\lambda f. (f = 1_x)): \prod_{x: C} \text{Hom}(x, x) \rightarrow \mathbf{Prop}_{\mathcal{U}}$$

where we know that $f = 1_x$ is a mere proposition since $\text{Hom}(x, x)$ is an h -set. Exactly analogously, since \circ ensures that each type $\text{Hom}(x, z)$ will be inhabited given $f: \text{Hom}(x, y)$ and $g: \text{Hom}(y, z)$ we define

$$\circ_C: \prod_{x,y,z: C} \text{Hom}(x, y) \rightarrow \text{Hom}(y, z) \rightarrow \text{Hom}(x, z) \rightarrow \mathbf{Prop}_{\mathcal{U}}$$

The verification of the axioms for this data is then entirely straightforward and we omit the details. So we can now write $q(C)$ for the \mathbb{T}_{cat} -model given by the data

$$(C, \text{Hom}, I_C, \circ_C, t_1^C, t_2^C, t_3^C, t_4^C, t_5^C, t_6^C, t_7^C)$$

It is now easy to check that p and q are quasi-inverses establishing the required equivalence. \square

A *strict category* ([Uni13], Definition 9.6.1) is a precategory in which the type of objects is an h -set. We write \mathbf{StrCat} for the type of strict categories. Let $\mathcal{L}_{\text{strcat}}$ be the signature whose category part is the same as \mathcal{L}_{cat} but with $d(O) = 1$. Let $\mathbb{T}_{\text{strcat}}$ be the $\mathcal{L}_{\text{strcat}}$ -theory that contains the same axioms as \mathbb{T}_{cat} together with the following axiom which expresses that O is an h -set:

$$(8) \quad \forall x, y: O. \exists p: x =_O^1 y. \forall q: x =_O^1 p. p =_O^2 q$$

We can now show that strict categories are “1-elementary.”

Corollary 7.2. $\mathbf{StrCat} \simeq \mathbf{Mod}(\mathbb{T}_{\text{strcat}})$

Proof. Given Proposition 7.1 it remains to check that the new axiom (8) for strict categories ensures that $O^{\mathcal{M}}$ is an h -set for any $\mathbb{T}_{\text{strcat}}$ -model \mathcal{M} . By Hedberg’s Theorem

([Uni13], Theorem 7.2.1) it suffices to prove that

$$(*) \quad \prod_{x: O^{\mathcal{M}}} \prod_{p: \text{Id}_{O^{\mathcal{M}}}(x,x)} p = \mathbf{refl}_x$$

is inhabited. Now suppose we are given an inhabitant of the (untruncated) translation of axiom (8)

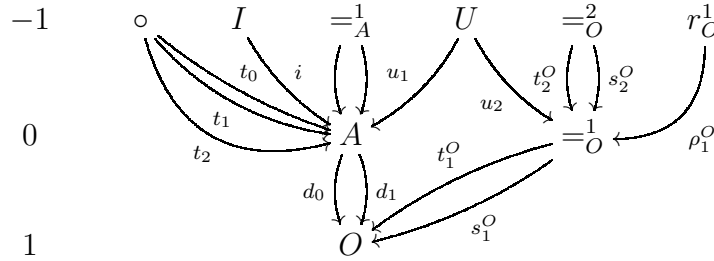
$$\eta: \prod_{x,y: O^{\mathcal{M}}} \prod_{p: \text{Id}_{O^{\mathcal{M}}}(x,y)} \sum_{q: \text{Id}_{O^{\mathcal{M}}}(x,y)} p = q$$

Setting $\eta_{x,x} \equiv \langle c, \theta \rangle$ and noting that $\theta_{\mathbf{refl}_x}$ is a proof that $c = \mathbf{refl}_x$ we get an inhabitant of $(*)$. But by the definition of our semantics we know that $\text{Id}_{O^{\mathcal{M}}}(x,x)$ is an h -set and therefore $(*)$ will be an h -prop. Therefore, the proper (truncated) translation of axiom (8) will also produce an inhabitant of $(*)$, and we are done. \square

More interestingly, we can show that univalent categories are also “1-elementary” although we have to expand our signature to do so. A *univalent category* ([Uni13], Definition 9.1.6) is a precategory satisfying the following additional datum, which expresses the fact that the canonical map $\mathbf{idtoiso}_{a,b}: a = b \rightarrow a \cong b$ is an equivalence for all a, b :

$$(7) \text{ cat}: \prod_{a,b: C} \mathbf{isequiv}(\mathbf{idtoiso}_{a,b})$$

We write **Unicat** for the type of univalent categories. Now let $\mathcal{L}_{\text{ucat}}$ be the following 1-signature



subject to all the same relations as \mathcal{L}_{cat} as well as the additional relation $t_1^O u_1 = s_1^O u_2$. We can then define \mathbb{T}_{ucat} as the $\mathcal{L}_{\text{ucat}}$ -theory given by the axioms of \mathbb{T}_{cat} together with the following extra axioms:

- (8) $\forall x, y: O. \forall f: A(x, y). \text{Iso}(f) \rightarrow (\exists! p: x =_O^1 y. U(f, p))$
- (9) $\forall x, y: O. \forall f: A(x, y). \forall p: x =_O^1 y. U(f, p) \rightarrow \text{Iso}(f)$
- (10) $\forall x: O. \forall f: A(x, x). \forall p: x =_O^1 x. (I(f, x) \wedge U(f, p) \rightarrow r_O^1(p, x))$
- (11) $\forall x, y: O. \forall f: A(x, y). \forall p, q: x =_O^1 y. (U(f, p) \wedge U(f, q) \rightarrow p =_O^2 q)$

where we have used the following abbreviations:

$$\text{Iso}(f) \equiv \exists g: A(x, y) \exists h_1: A(x, x) \exists h_2: A(y, y).$$

$$\circ(f, g, h_1) \wedge \circ(g, f, h_2) \wedge I(h_1) \wedge I(h_2)$$

$$\exists! p: x =_O^1 y. U(f, p) \equiv \exists p: x =_O^1 y. (U(f, p) \wedge (\forall q: x =_O^1 y. (U(f, q) \rightarrow p =_O^2 q)))$$

Thus, axioms (8)-(10) express that U is a bijective relation between isomorphisms and paths that sends identity to reflexivity and axiom (11) expresses that U a functional relation. We now obtain the following.

Proposition 7.3. $\mathbf{UniCat} \simeq \mathbf{Mod}(\mathbb{T}_{\text{ucat}})$

Proof. From Proposition 7.1 we can assume that the data for a model of \mathbb{T}_{ucat} is given by the same data as that of for a precategory, together with the interpretation of U . Given AUC, for any given \mathbb{T}_{ucat} -model \mathcal{M} we can extract from $U^{\mathcal{M}}$ a section

$$u^{\mathcal{M}}: \prod_{x,y: O^{\mathcal{M}}} \text{Id}_{O^{\mathcal{M}}}(x, y) \rightarrow \text{Iso}^{\mathcal{M}}(x, y)$$

where

$$\text{Iso}^{\mathcal{M}}(x, y) \equiv \prod_{f: A^{\mathcal{M}}(x, y)} \text{isiso}(f)$$

such that

$$\pi: \prod_{x,y: O^{\mathcal{M}}} \text{isequiv}(u_{x,y})$$

Thus we get

$$\mathbf{Mod}(\mathbb{T}_{\text{ucat}}) \simeq \sum_{\substack{C: \mathbf{Precat} \\ u: \prod_{x,y: O^C} \text{Id}_{O^C}(x,y) \rightarrow \text{Iso}^C(x,y)}} \prod_{x,y: O^C} \text{isequiv}(u_{x,y})$$

We can now take \mathbf{Unicat} to be the type

$$\sum_{C: \mathbf{Precat}} \prod_{x,y: O^C} \text{isequiv}(\text{idtoiso}_{x,y})$$

There is then a natural map f from \mathbf{UniCat} to $\mathbf{Mod}(\mathbb{T}_{\text{ucat}})$ which sends $\langle D, \text{univ} \rangle$ to $\langle D, \text{idtoiso}, \text{univ} \rangle$. For a given

$$\langle D, u, \pi \rangle: \mathbf{Mod}(\mathbb{T}_{\text{ucat}})$$

the homotopy fiber of f over $\langle D, u, \pi \rangle$ is given by

$$\text{hfib}_f(\langle D, u, \pi \rangle) \equiv \sum_{\langle C, \text{univ} \rangle} \langle C, \text{idtoiso}, \text{univ} \rangle = \langle D, u, \pi \rangle$$

To show that $\text{hfib}_f(\langle D, u, \pi \rangle)$ is contractible it clearly suffices to show that for all $x, y: O$, $u_{x,y} = \text{idtoiso}_{x,y}$ and by function extensionality this reduces to giving an inhabitant of

$$\prod_{x,y,p} u_{x,y}(p) = \text{idtoiso}_{x,y}(p)$$

which by path induction reduces to providing an inhabitant of

$$\prod_x u_{x,x}(\text{refl}_x) = \text{idtoiso}_{x,x}(\text{refl}_x)$$

But by the axioms of univalent categories and of \mathbb{T}_{ucat} we get that both sides of the equation are (propositionally) equal to the (unique) identity map on x . Thus f is an equivalence and we are done. \square

Remark 7.4. Clearly the axioms of \mathbb{T}_{ucat} employ full first-order logic. It is unclear whether \mathbf{Unicat} can also be axiomatized in the coherent fragment over $\mathcal{L}_{\text{ucat}}$.

Proposition 7.3 illustrates the kind of result that n -logic was designed to tackle, and which justifies the title of this paper. Namely, we want to tackle traditional model-theoretic questions (e.g. of axiomatizability) but about structures defined on homotopy types rather than sets. From this point of view, many future projects and questions suggest themselves. We list a few:

- (1) Extending the syntax of n -logic to the case $n = \infty$ in such a way as to make it possible to define semantics as in Definition 3.2. (This is related to the well-known open problem of managing infinite chains of coherence data in HoTT.)
- (2) Proving a general completeness theorem for all finite $n > 1$. (Completeness in the case of $n = 1$ will be proven in the sequel, for both homotopy and set-theoretic semantics.)
- (3) Characterizing n -elementary types in general, i.e. proving a Loś Theorem for n -logic along the lines of [CK90], Theorem 4.1.12. (This is likely to be non-trivial even in the case of $n = 1$.)

APPENDIX: PROOF SYSTEM

We present the “old” rules of the proof system \mathcal{D} of Section 5 to which the “new” rules (Eq-intro), (J) and (R) are added. We fix a signature \mathcal{L} and we assume that every formula that appears below is a well-formed \mathcal{L} -formula and that every sequent is well-formed. Our presentation combines elements from [Jac99, Mak95, Joh03]. As usual, the double lines denote a rule that goes in either direction.

Structural Rules

$$\begin{array}{c}
\overline{\Gamma \mid \phi \vdash \phi} \quad (\text{iden}) \\
(\text{Sub}) \frac{\Delta \mid s(\phi) \vdash s(\psi)}{\Gamma \mid \phi \vdash \psi} \quad s: \Gamma \Rightarrow \Delta \\
\frac{\Gamma \mid \phi \vdash \psi \quad \Gamma \mid \psi \vdash \chi}{\Gamma \mid \phi \vdash \chi} \quad (\text{Cut}) \quad \frac{\Gamma \mid \phi \vdash \psi}{\Gamma, x: K \mid \phi \vdash \psi} \quad (\text{Con-wk}) \\
(\text{Con-exch}) \frac{\Gamma, y: K', x: K, \Gamma' \mid \phi \vdash \psi}{\Gamma, x: K, y: K', \Gamma' \mid \phi \vdash \psi} \quad x \notin \text{dep}(y)
\end{array}$$

Logical Rules

$$\begin{array}{c}
\overline{\Gamma \mid \phi \vdash (\top)} \quad \top \quad \overline{\Gamma \mid \perp \vdash \phi} \quad (\perp) \\
(\wedge) \frac{\Gamma \mid \theta \vdash \phi \quad \Gamma \mid \theta \vdash \psi}{\Gamma \mid \theta \vdash \phi \wedge \psi} \quad (\vee) \frac{\Gamma \mid \phi \vdash \theta \quad \Gamma \mid \psi \vdash \theta}{\Gamma \mid \phi \vee \psi \vdash \theta} \\
(\rightarrow) \frac{\Gamma \mid \theta \wedge \phi \vdash \phi}{\Gamma \mid \theta \vdash \phi \rightarrow \psi} \\
(\forall) \frac{\Gamma, x: K \mid \theta \vdash \phi}{\Gamma \mid \forall x: K. \phi} \quad x \notin |\Gamma| \quad (\exists) \frac{\Gamma, x: K \mid \phi \vdash \theta}{\Gamma \mid \exists x: K. \phi \vdash \theta} \quad x \notin |\Gamma|
\end{array}$$

If we are working in the coherent fragment then we also add the following two rules, which are otherwise derivable:

$$\overline{\Gamma \mid \phi \wedge (\exists x \psi) \vdash \exists x(\phi \wedge \psi)} \quad (\text{Frob}), x \notin \text{FV}(\phi)$$

$$\overline{\Gamma \mid \phi \wedge (\psi \vee \chi) \vdash (\phi \wedge \psi) \vee (\phi \wedge \chi)} \quad (\text{Dist})$$

Finally, to get \mathcal{D}^{cl} we add the law of the excluded middle:

$$\overline{\Gamma \mid \top \vdash \phi \vee (\phi \rightarrow \perp)} \quad (\text{LEM})$$

REFERENCES

- [ANS14] B. Ahrens, P. North, and M. Shulman, *Univalent FOLDS* (2014). Abstract available at: http://www.ams.org/amsmtgs/2223_abstracts/1103-18-196.pdf.
- [Car86] J. Cartmell, *Generalized algebraic theories and contextual categories*, Annals of Pure and Applied Logic **32** (1986), 209–243, available at <http://www.sciencedirect.com/science/article/pii/0168007286900539>.
- [CCHM15] C. Cohen, T. Coquand, S. Huber, and A. Mörtberg, *Cubical type theory: a constructive interpretation of the univalence axiom*, 2015.
- [CK90] C. C. Chang and H. Jerome Keisler, *Model theory*, Elsevier, 1990.
- [Hof97] Martin Hofmann, *Syntax and semantics of dependent types*, Extensional constructs in intensional type theory, 1997, pp. 13–54.
- [HS98] M. Hofmann and T. Streicher, *The groupoid interpretation of type theory* **36** (1998), 83–111. MR1686862
- [Jac99] B. Jacobs, *Categorical logic and type theory*, Elsevier, 1999.
- [Joh03] P. Johnstone, *Sketches of an elephant: A topos theory compendium*, Oxford University Press, Oxford, 2003.
- [KLV14] K. Kapulkin, P. Lumsdaine, and V. Voevodsky, *The simplicial model of univalent foundations*, arXiv:1211.2851v2 (2014).
- [Mak95] M. Makkai, *First order logic with dependent sorts with applications to category theory*, 1995. <http://www.math.mcgill.ca/makkai/folds/foldsinpdf/FOLDS.pdf>.
- [ML84] Per Martin-Löf, *Intuitionistic type theory*, Bibliopolis, 1984.
- [ML98] Saunders Mac Lane, *Categories for the working mathematician (graduate texts in mathematics)*, Springer, 1998.
- [PW14] E. Palmgren and O. Wilander, *Constructing categories and setoids of setoids in type theory*, arXiv preprint arXiv:1408.1364 (2014).
- [Uni13] Univalent Foundations Program, *Homotopy type theory: Univalent foundations of mathematics* (2013). <http://homotopytypetheory.org/book>.
- [VAG⁺] Vladimir Voevodsky, Benedikt Ahrens, Daniel Grayson, et al., UniMath: *Univalent Mathematics*. Available at <https://github.com/UniMath>.
- [Voe14] V. Voevodsky, *Subsystems and regular quotients of C-systems*, arXiv 1406.5389, submitted (2014), 1–11.

DEPARTMENT OF PHILOSOPHY, PRINCETON UNIVERSITY
 Current address: 1879 Hall, Princeton, NJ 08544, USA
 E-mail address: dtsement@princeton.edu